

UNIVERSIDAD CARLOS III DE MADRID

PROYECTO FIN DE CARRERA

Ingeniería Técnica en Informática de Gestión



APLICACIÓN PARA LA DESCARGA DE NOTICIAS EN DISPOSITIVOS MÓVILES

Autor: Natalia Rodríguez González

Tutor: Israel González Carrasco

Leganés, Enero de 2016

Título: Aplicación para la descarga de noticias en dispositivos móviles

Autor: Natalia Rodríguez González

Tutor: Israel González Carrasco

EL TRIBUNAL

Presidente: _____

Vocal: _____

Secretario: _____

Realizado el acto de defensa y lectura del Proyecto Fin de Carrera el día __ de _____ de 20__ en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la CALIFICACIÓN de

VOCAL

SECRETARIO

PRESIDENTE

AGRADECIMIENTOS

Por fin llegó el momento. Hace mucho tiempo que llevo guardado dentro este pedacito de PFC, antes de empezar ya tenía claro lo que en este apartado quedaría grabado.

Muchas gracias Israel, por tu tiempo y dedicación, por ayudarme ha pasar por la línea de meta de mi carrera académica y personal más importante.

Por supuesto, tengo que mencionaros a vosotros: Aceves, Alfon, Burrita, Facio, Patita, Vitorino, en especial mi compañero y amigo de prácticas David (porque hay equipos, que son algo más que un equipo), soy una afortunada por encontrarme, como siempre he dicho, con lo mejor que ha pasado y pasará por la UC3M, sin vosotros nada hubiera sido igual, no olvido cada minuto juntos.

Agradezco a mi familia su apoyo y cariño constante, por hacer transparentes los kilómetros que nos separan, lo nuestro sí que es trabajo en equipo. Somos como una máquina de engranajes perfectamente encajados, con un eje impulsor, que aunque han dejado de estar con nosotros, son el motor y esencia de nuestras vidas.

Gracias Mónica (prima y amiga), ya son más de veinte años de momentos congelados en recuerdos que jamás se olvidarán. Y sin hablar, sólo al mirar sabemos llegar a entender que jamás ni nada ni nadie en la vida nos separarán [1]. Tus mensajes de ánimos diarios de los últimos meses han sido una fuente de energía para aguantar el ritmo y no flaquear.

No puedo olvidarme de la persona que forjó la base sobre la que se ha construido la torre de esta ingeniería, gracias a mi tía Begoña, por las tardes corriéndome deberes, preguntado lecciones e inculcándome la importancia de libros y los estudios, entre otras muchas cosas. Aquí está el resultado de tu costoso y persistente esfuerzo.

Como todo buen banquete termina con un dulce postre, mi agradecimiento más especial, no sólo porque mi paso por la UC3M ha sido gracias a ella, su apoyo, cariño y esfuerzo, sino porque lo soy todo gracias a ella, porque ella sola es cómo el ejecito de la película 300 [2], porque como le digo cada día, "es la más mejor madre del mundo mundial", **gracias mamá**.

*"CAMIÑANDO FAISE O CAMIÑO
E CAMIÑANDO IMOS CHEGAR"*

JUAN PARDO – ALMA GALLEGA

RESUMEN

En la actualidad vivimos en una sociedad rodeada de información y caracterizada por la comunicación a través de nuevas tecnologías. Cada vez la sociedad demanda y consume más información, exigiendo que este actualizada y con unos contenidos específicos.

Por esta razón, van surgiendo nuevas tecnologías de la Información (TIC) que facilitan y estandarizan el acceso a la información.

Estas tecnologías permiten construir una Sociedad de la información basada en tecnologías que facilitan el acceso a Información, debido a su rapidez y flexibilidad.

Dentro de estas nuevas innovaciones surge RSS, un estándar basado en lenguaje XML para compartir y publicar información. RSS permite tener información actualizada sobre aquellos contenidos favoritos a los cuales se subscriben cada usuario, sin necesidad de que tenga que visitar cada página web.

RSS se basa en un mecanismo sencillo, por el cual los medios distribuyen información (feed) en unas fuentes de contenidos, a los cuales los usuarios pueden suscribirse y recopilarlos en un lector o agregador RSS.

El objetivo de este PFC es implementar RSS para desarrollar un lector que permita suscribirse a enlaces de cualquier medio y gestionarlos de manera sencilla, permitiendo tener en un solo click toda la información actualizada.

Este lector se basa en una aplicación móvil llamada "FeedBack" disponible para dispositivos móviles con sistema operativo iOS. Mediante esta app, el usuario podrá suscribirse y gestionar fácilmente los diferentes RSS de su interés y tener al alcance toda la información actualizada que precise.

Palabras clave: información actualizada, tecnologías de la información, lector, RSS, aplicación móvil.

ABSTRACT

We are currently living in a society surrounded by information and characterized by communication through new technologies. Each time, society demands and consumes more information, requiring that it is up to date and with a specific content.

Therefore, new information technologies (ICTs) are emerging that facilitate and standardize access to information.

These technologies allow to construct a society based on technologies that facilitate access to information, due to its simplicity and flexibility.

Within these new innovations appears RSS, an XML-based language to share and publish information. RSS provide updated information on those favorites content which each user has subscribed, avoiding visiting each web page.

RSS is based on a simple mechanism by which the mass media disclose information (feed) in some sources of content, where users can subscribe and compile them into a RSS reader or aggregator.

The aim of this PFC is to implement and develop an RSS reader that allows you to subscribe to any media link and manage them easily, allowing all the updated information by only one-click.

This reader is based on a mobile application called "Feedback" available for mobile devices with IOS operating system. Through this app, users can subscribe and manage different RSS of their interest and have available all the updated information that users need.

Keywords: updated information, information technologies, reader, RSS, mobile application.

ÍNDICE GENERAL

AGRADECIMIENTOS	4
RESUMEN	6
ABSTRACT	7
ÍNDICE GENERAL	9
ÍNDICE DE FIGURAS	12
ÍNDICE DE TABLAS	14
CAPÍTULO 1: INTRODUCCIÓN	16
1.1. CONTEXTO	16
1.2. PROBLEMA	17
1.3. MOTIVACIÓN	18
1.4. OBJETIVOS	19
1.5. MÉTODO DE RESOLUCIÓN	19
1.6. GLOSARIO DE TÉRMINOS Y ABREVIATURAS	21
1.6.1. TÉRMINOS	21
1.6.2. ABREVIATURAS	23
1.7. ESTRUCTURA DE LA MEMORIA	24
CAPÍTULO 2: ESTADO DEL ARTE	26
2.1. INTRODUCCIÓN A LOS LECTORES RSS	26
2.2. ¿QUÉ ES UN CANAL/SERVICIO RSS?	27
2.3. ESTRUCTURA DE UN CANAL RSS	27
2.3.1. ELEMENTOS DEL NODO CHANEL	28
2.3.2. ELEMENTOS DE LOS NODOS ITEM	28
2.4. VERSIONES DE RSS	29
2.5. ¿CÓMO SABER SI UN SITIO WEB OFRECE SINDICACIÓN?	30
2.6. ANÁLISIS DE APLICACIONES SIMILARES	30
2.6.1. FEEDLY	31
2.6.2. FEEDDEMON	31
2.6.3. FLIPBOARD	32
2.6.4. FUSION	33
2.6.5. NETVIBES	33

2.6.6.	LINKEDIN PULSE	34
2.7.	ENTORNO DE TRABAJO SELECCIONADO	35
2.8.	ENTORNO DE DESARROLLO IOS	36
2.8.1.	ARQUITECTURA	37
2.8.2.	LENGUAJE DE PROGRAMACIÓN OBJECTIVE-C	38
2.9.	PATRONES DE DISEÑO UTILIZADOS	40
CAPÍTULO 3: ESPECIFICACIÓN DE REQUISITOS		47
3.1.	INTRODUCCIÓN	47
3.2.	DESCIPCIÓN GENERAL	47
3.2.6.1.	REQUISITOS FUNCIONALES	50
3.2.6.2.	REQUISITOS NO FUNCIONALES	53
3.3.	ESPECIFICACIÓN DE CASOS DE USO	54
CAPITULO 4: DISEÑO DE LA SOLUCIÓN		63
4.1.	INTRODUCCIÓN	63
4.2.	DIAGRAMAS DE SECUENCIA	63
4.2.1.1.	ALTA ENLACE RSS	64
4.2.1.2.	BORRAR ENLACE RSS	65
4.2.1.3.	MOSTRAR ENLACE RSS	66
4.3.	DIAGRAMAS DE CLASES	74
CAPITULO 5: PRUEBAS		78
5.1.	INTRODUCCIÓN	78
5.2.	PRUEBAS INTEGRADAS	78
5.3.	TRAZABILIDAD CASOS DE PRUEBA-REQUISITOS FUNCIONALES	86
CAPITULO 6: PRESUPUESTO		87
6.1.	INTRODUCCIÓN	87
6.2.	COSTES DE PERSONAL	88
6.3.	COSTES MATERIALES	88
6.4.	OTROS COSTES DIRECTOS	89
6.5.	COSTE TOTAL	90
6.6.	DURACIÓN DEL PROYECTO	91
CAPITULO 7: CONCLUSIONES Y LINEAS FUTURAS		92
7.1.	CONCLUSIONES	92
7.2.	LINEAS FUTURAS	93
REFERENCIAS		95

ANEXO I: MANUAL DE USUARIO	97
A.1. INTRODUCCIÓN A FEEDBACK.....	97
A.2. CÓMO SUSCRIBIRSE A UN SERVICIO RSS	98
A.3. CÓMO VISUALIZAR LOS FEEDS.....	104
A.4. CÓMO ACTUALIZAR LOS FEEDS.....	107
A.5. CÓMO ARCHIVAR FEEDS.....	108
A.6. CÓMO VISUALIZAR FEEDS ARCHIVADOS.	109
A.7. VISUALIZAR MENÚ	112

ÍNDICE DE FIGURAS

Ilustración 1: Ejemplo XML fuente RSS 2.0.....	28
Ilustración 2: Versiones RSS	29
Ilustración 3: Icono RSS	30
Ilustración 4: Logo FEEDLY.....	31
Ilustración 5: Logo FeedDemon	32
Ilustración 6: Logo FUSION	33
Ilustración 7: Logo NETVIBES.....	34
Ilustración 8: Logo LINKEDIN PULSE.....	34
Ilustración 9: Sistemas operativos móviles más utilizados en 2015	35
Ilustración 10: Sistemas operativos móviles más usados entre 2014 y 2015	35
Ilustración 11: Logo APPLE & ANDROID	36
Ilustración 12: Arquitectura IOS	37
Ilustración 13: Logo X-CODE	38
Ilustración 14: Logo SQLITE	39
Ilustración 15: Diferencia entre un sistema gestor de base de datos tradicional y SQLITE	40
Ilustración 16: Patrón MODELO-VISTA-CONTROLADOR.....	41
Ilustración 17: Delegate de funcionalidad para una ventana	42
Ilustración 18: Aproximación sin centro de notificaciones	44
Ilustración 19: Uso del centro de notificaciones.....	45
Ilustración 20: Patrón de diseño Target - Action.....	46
Ilustración 21: Diagrama de casos de uso.....	55
Ilustración 22: Diagrama de secuencia alta enlace RSS.....	64
Ilustración 23: Diagrama de secuencia baja enlace RSS	65
Ilustración 24: diagrama de secuencia mostrar enlace RSS	66
Ilustración 25: Diagrama de secuencia listar feeds por RSS.....	67
Ilustración 26: Diagrama de secuencia visualizar feed.....	68
Ilustración 27: Diagrama de secuencia actualizar feeds.....	69
Ilustración 28: Diagrama de secuencia listar feeds archivados.....	70
Ilustración 29: Diagrama de secuencia borrar feed archivado.....	71
Ilustración 30: Diagrama de secuencia archivar feed	72
Ilustración 31: Diagrama de secuencia visualizar feed en sitio web	73
Ilustración 32: Diagrama de clases	74
Ilustración 33: Diagrama arquitectura de clases	76
Ilustración 34: Maqueta interfaz I.....	77
Ilustración 35: Maqueta interfaz II	77
Ilustración 36: Presupuesto aplicación.....	90
Ilustración 37: Diagrama Gantt	91
Ilustración 38: Icono RSS II	97
Ilustración 39: Identificación enlaces RSS 20minutos.es	98
Ilustración 40: Identificación enlaces RSS boletín oficial UC3M	98
Ilustración 41: URL RSS 20minutos.es.....	99

Ilustración 42: URL RSS boletín oficial UC3M.....	99
Ilustración 43: Pantalla inicio FeeDBacK.....	100
Ilustración 44: Ventana FeeDBacK gestión enlaces RSS.....	101
Ilustración 45: Datos obligatorios alta RSS.....	101
Ilustración 46: Operación alta servicio RSS.....	102
Ilustración 47: Alta servicio RSS realizada.....	103
Ilustración 48: Error enlace RSS versión no soportada.....	104
Ilustración 49: Listado feeds por RSS.....	105
Ilustración 50: Ventana listado/visualización feeds por RSS.....	105
Ilustración 51: Ventana visualizar feed.....	106
Ilustración 52: Botón para ver feed en sitio web.....	107
Ilustración 53: Actualizar feeds.....	108
Ilustración 54: Botón para archivar feed.....	109
Ilustración 55: Listado feeds archivados.....	110
Ilustración 56: Ventana listado/visualización feeds archivados.....	111
Ilustración 57: Borrar feed archivado.....	112
Ilustración 58: Visualizar menú principal.....	113

ÍNDICE DE TABLAS

Tabla 1: Términos	22
Tabla 2: Abreviaturas.....	23
Tabla 3: Ejemplo para especificación de requisitos.....	49
Tabla 4: RF-01 Suscribirse a un RSS	50
Tabla 5: RF-02 Borrar RSS	50
Tabla 6: RF-03 Validar RSS.....	51
Tabla 7: RF-04 Listar RSS suscritos	51
Tabla 8: RF-05 Listar feeds por RSS.....	51
Tabla 9: RF-06 Visualizar feed	51
Tabla 10: RF-07 Visualizar feed en sitio web.....	52
Tabla 11: RF-08 Archivar feed	52
Tabla 12: RF-09 Actualizar feeds	52
Tabla 13: RF-10 Listar feeds archivados.....	52
Tabla 14: RF-11 Marcar feed como no leído	52
Tabla 15: RNF-01 Dispositivos soportados	53
Tabla 16: RNF-02 Versión mínima soportada	53
Tabla 17: RNF-03 Uso patrón Pull to Refresh.....	53
Tabla 18: RNF-04 Accesible a través de internet.....	53
Tabla 19: RNF-05 Cambio de orientación	54
Tabla 20: CU-01 Gestionar RSS	56
Tabla 21: CU-02 Listar feeds por RSS.....	57
Tabla 22: CU-03 Visualizar feed	57
Tabla 23: CU-04 Actualizar feeds	58
Tabla 24: CU-05 Listar feeds archivados	59
Tabla 25: CU-06 Marcar feed como no leído.....	60
Tabla 26: CU-07 Archivar feed	60
Tabla 27: CU-08 Visualizar feed en sitio web	61
Tabla 28: Trazabilidad requisitos-casos de uso.....	62
Tabla 29: PI-01 Suscripción a un enlace RSS	79
Tabla 30: PI-02 Borrado de un enlace RSS.....	80
Tabla 31: PI-03 Validar servicio RSS a suscribir	80
Tabla 32: PI-04 Validar servicio RSS a suscribir	81
Tabla 33: PI-05 Listar RSS suscritos	81
Tabla 34: PI-06 Listar feeds a por RSS.....	82
Tabla 35: PI-07 Visualizar feed	82
Tabla 36: PI-08 Visualizar feed en sitio web.....	83
Tabla 37: PI-09 Archivar feed	84
Tabla 38: PI-10 Actualizar feeds	84
Tabla 39: PI-11 Listar feeds archivados.....	85
Tabla 40: PI-12 Marcar feed como no leído	86
Tabla 41: Trazabilidad casos de prueba - requisitos funcionales	86

Tabla 42: Costes de personal88

Tabla 43: Costes materiales89

Tabla 44: Otros costes directos.....89



CAPÍTULO 1: INTRODUCCIÓN

El presente documento toma como guía la plantilla de memoria de PFC proporcionada por la UC3M.

En este primer capítulo se va a dar una visión introductoria del proyecto, exponiendo el problema que se pretende resolver, la motivación y los objetivos del mismo, así como el hardware y software utilizado para su ejecución. Además, se explicará la metodología usada para la resolución del problema y se dará un enfoque general de la estructura del contenido de esta memoria.

Además, se expone una descripción sobre el contenido de los siguientes apartados, así como un glosario completo de términos y abreviaturas utilizadas a lo largo de todo el documento, para facilitar la comprensión del mismo al lector.

1.1. CONTEXTO

El nuevo contexto de la Sociedad de la Información y el desarrollo de las Tecnologías de la Información y la Comunicación, han provocado que actualmente exista una gran cantidad de información accesible al usuario, mediante distintas tecnologías y dispositivos.

Internet es sin duda la tecnología decisiva en esta sociedad, por la cual los ciudadanos pueden acceder a esa información y publicar nuevos contenidos. Esto provoca que actualmente Internet se haya convertido en un espacio donde existe una cantidad inmensa de información e innumerables posibilidades de acceder a ella.

Esta gran cantidad de información permite que cualquier usuario con acceso a la red y un dispositivo para acceder a ella (Smartphone, PC, Tablet, etc.), pueda estar informado de aquellos contenidos que le interese, accediendo a una gran cantidad de sitios con contenidos gratuitos.

Cada día se crean nuevos medios de comunicación y se actualizan millones de páginas con nuevos contenidos en Internet. No obstante, esto provoca que sea una tarea complicada para el usuario gestionar esta información y mantenerse al día con aquellas noticias de su interés.

Por esta razón, surge la necesidad de una aplicación como "FeedBack" que permite gestionar toda la información atractiva para el usuario de una manera sencilla utilizando el estándar de comunicaciones RSS. En este primer capítulo, expondremos la problemática que pretende solventar la aplicación y los principales objetivos que le aplican.

1.2. PROBLEMA

Actualmente existen muchos usuarios que consumen un gran volumen de información y para ello deben acceder a múltiples páginas web cuyo contenido cambia constantemente y en cualquier momento. Ejemplos de este tipo de sitios web son portales de noticias, páginas con información acerca de productos, blogs, etc.

Por esta razón, el usuario se ve obligado a comprobar constantemente sus sitios web preferidos para chequear si existe nuevo contenido. Esto se convierte en una tarea tediosa que requiere acceder a cada una de las páginas, cargar la información, recordar su formato y modo de navegación y encontrar donde se quedó la última vez o detectar qué información ha sido actualizada. Todo esto implica que la búsqueda de información sea una actividad lenta y requiera que el usuario deba invertir más tiempo de navegación.

Una solución inicial a este problema fue la notificación por correo electrónico con las publicaciones de nuevo contenido en las páginas donde el usuario se hubiese suscrito por correo electrónico.

No obstante, cuando se reciben las notificaciones de correo electrónico de varios sitios web, se suelen recibir de manera desorganizada, y puede llegar a ser molesto, ya que se pueden recibir gran cantidad de noticias, y además recibir SPAM u otra información no deseada.

Además, en las suscripciones vía correo electrónico a contenidos de información provocan que cualquier cancelación en la suscripción resulte dificultosa ya que en algunos medios tienes que especificar las razones por las que quiere darse de baja para confirmar la cancelación del servicio.

1.3. MOTIVACIÓN

Como alternativa a lo anteriormente expuesto, nacen los servicios RSS (Really Simple Syndication), para difundir o compartir contenido en la web. RSS es una forma muy sencilla para que puedas recibir en tu dispositivo (Smartphone, Tablet, etc.) la información actualizada sobre tus páginas web favoritas, sin necesidad de que tengas que visitarlas una a una.

El mecanismo de funcionamiento de RSS consiste en un sitio web que publica información y mantiene una lista de notificaciones que se van actualizando en su página web. Esta lista de notificaciones se denomina "RSS Feed". Las personas que estén interesadas en conocer las últimas noticias o cambios pueden suscribirse a esta lista, mediante un lector de RSS. Estos lectores o agregadores de "RSS Feed" son básicamente un software que accede automáticamente a los feeds de estos sitios web a los cuales el usuario se ha suscrito y permiten visualizar las noticias actualizadas y organizar los resultados.

Este proyecto surge como respuesta a la necesidad de los usuarios de disponer de un lugar centralizado que les permita mantenerse informados sin necesidad de visitar cada una de las páginas web para ver si han añadido o no algún artículo que les pueda interesar. Además, les permitirá ahorrar tiempo de navegación mediante el uso de un lector RSS; y no será necesario proporcionar un correo electrónico, evitando problemas de correo no deseado.

RSS por tanto facilita de una manera notable el acceso al usuario a aquellas novedades y actualizaciones de sus enlaces favoritos de manera centralizada, y que sea el propio usuario quien gestione estas suscripciones, ya que es él quien decide donde suscribirse y cuando cancelarlo sin ninguna dificultad.

Además, los servicios RSS son generalmente de carácter gratuito y no generan Spam, ya que no es necesario aportar ningún correo ni cuenta para su suscripción.

En modo de resumen, mediante un lector RSS el usuario estará automáticamente informado sobre todas las novedades que se han producido en todas las páginas web que hayan dado de alta de una forma sencilla y ágil.

1.4. OBJETIVOS

El objetivo principal de este proyecto consiste en diseñar y desarrollar una herramienta que permita suscribirse a servicios aquellos RSS proporcionados por multitud de sitios web, y poder visualizar la información actualizada publicada por éstos, de forma reunida y categorizada en un mismo lugar y a un solo clic de distancia.

En base a este objetivo principal, se proponen los siguientes objetivos parciales:

- Diseñar y desarrollar una aplicación con un interfaz amigable y de sencilla usabilidad, de forma que sea transparente para el usuario la tecnología utilizada y este pueda abstraerse de manipular la información.
- Debe poder ser accedida desde dispositivos iPad y su presentación debe estar optimizada para este tipo de elementos.
- Facilitar un sistema de suscripción a los servicios RSS de los sitios web favoritos de los usuarios. Debe permitir enlazar la aplicación con los servicios RSS suscritos de tal forma que el usuario pueda recibir la información actualizada en cada momento.
- Proporcionar la funcionalidad necesaria para que los usuarios puedan gestionar los "RSS Feed", como modificar y anular sus suscripciones en cualquier momento.
- Ofrecer los medios necesarios para poder visualizar el contenido actualizado por los servicios RSS suscritos previamente.

1.5. MÉTODO DE RESOLUCIÓN

El diseño y desarrollo del lector RSS "FeedBack" está basado en metodologías ágiles en vez de un modelo predictivo. La elección de este método se basa en que el desarrollo de esta aplicación requería un alto grado de cambios y nuevas funcionalidades, por lo que una planificación estricta podría bloquear la evolución del proyecto. Por otra parte, la metodología ágil permite un mayor solapamiento en las distintas fases del proyecto.

Otro motivo para la elección de este tipo de metodologías se basa en que se crean puntos de control no tan rígidos como en metodologías predictivas, lo que permite realizar un seguimiento del proyecto con el tutor adecuado sin acotar la creatividad ni limita la solución.

Se ha optado por un enfoque iterativo e incremental basado en dividir el proyecto en bloques con una planificación aproximada de 2-4 semanas cada uno, llamadas iteraciones. Estas iteraciones consisten en realizar una parte del proyecto de aquellos casos de uso que comparten una funcionalidad o usabilidad, con todas sus fases incluidas. Estas fases incluyen los siguientes hitos:

- Planificación.
- Análisis de los requisitos.
- Diseño de la solución funcional y técnicamente.
- Desarrollo.
- Pruebas.
- Documentación en la memoria.

De este modo, se implementa un bloque completo que cumple las funcionalidades requeridas. En cada iteración se desarrollan nuevos bloques y se evolucionan aquellos ya implementados a partir de resultados de iteraciones anteriores. Esto permite un desarrollo flexible que se adapta a cualquier nueva funcionalidad evitando bloqueos o rehacer tareas.

El primer punto de partida fue la priorización de los requisitos más críticos y prioritarios para el desarrollo del lector RSS. En base a esta priorización, se han creado iteraciones con aquellos casos de uso con un comportamiento similar.

Cada iteración nueva requería revisar los requisitos de iteraciones anteriores, para revisar si debían modificarse y por tanto, incrementar su funcionalidad. En base a este análisis previo, se desarrollaban las iteraciones con los pasos anteriores en plazos de 2-4 semanas.

Hay que destacar que previo al desarrollo del proyecto, se ha requerido una fase de aprendizaje de la tecnología iOS y plataformas de desarrollo necesarios. Para tal fin, ha sido necesario realizar un curso online de desarrollo y revisar diversos manuales para conocer y entender la tecnología, de manera que pudiese servir para implementar la mejor solución posible.

1.6. GLOSARIO DE TÉRMINOS Y ABREVIATURAS

En este apartado se exponen las definiciones de términos y abreviaturas para el correcto entendimiento del presente documento.

1.6.1. TÉRMINOS

TÉRMINO	DEFINICIÓN
Android	Sistema operativo basado en el núcleo Linux. Fue diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes, tablets o tabléfonos; y también para relojes inteligentes, televisores y automóviles. Inicialmente fue desarrollado por Android Inc.
APP	Aplicación informática diseñada para ser ejecutada en teléfonos inteligentes, tabletas y otros dispositivos móviles.
Blackberry	Marca de teléfonos inteligentes desarrollada por la compañía canadiense BlackBerry
Blog	Sitio web que incluye, a modo de diario personal de su autor o autores, contenidos de su interés, actualizados con frecuencia y a menudo comentados por los lectores.
Chrome	Navegador web desarrollado por Google
Cocoa	Framework que permite el desarrollo de aplicaciones nativas para Mac OS X
e-book	Versión electrónica o digital de un libro
FEED	Término utilizado para dar nombre a un sistema de difusión de una fuente web o contenido web. Los Feeds se utilizan para suministrar información actualizada a los usuarios suscriptores a este medio por intermedio de un programa o aplicación.
Framework	Conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.
iOS	Sistema operativo móvil de la multinacional Apple Inc. Originalmente desarrollado para el iPhone (iPhone OS), después se ha usado en dispositivos como el iPod touch y el iPad.
iPAD	Línea de tabletas diseñadas y comercializadas por Apple Inc.
iPhone	Línea de teléfonos inteligentes diseñada y comercializada por Apple Inc.
iPod Touch	Reproductor multimedia, PDA, videoconsola portátil y plataforma móvil Wi-Fi diseñado y distribuido por Apple Inc.

Kindle Fire	Tableta diseñada por Amazon.com como la versión multimedia del lector de libros electrónicos
MAC OS	Sistema operativo creado por Apple para su línea de computadoras Macintosh
Maemo	Plataforma de desarrollo para dispositivos handheld basado en debian GNU/Linux
NOOK	Lector de libros electrónicos desarrollado por la empresa Barnes & Noble
Objective-C	Lenguaje de programación orientado a objetos creado como un superconjunto de C para que implementase un modelo de objetos parecido al de Smalltalk
Pull to refresh	Mecanismo estándar en aplicaciones móviles para actualizar el contenido de un listado o grid de datos con los elementos más recientes los cuales son añadidos en la parte superior del listado.
Smalltalk	Lenguaje reflexivo de programación, orientado a objetos
Smartphone	Teléfono inteligente sobre una plataforma informática móvil, con mayor capacidad de almacenar datos y realizar actividades, semejante a la de una minicomputadora, y con una mayor conectividad que un teléfono móvil convencional.
SPAM	Mensajes no solicitados, no deseados o con remitente no conocido (correo anónimo), habitualmente de tipo publicitario, generalmente enviados en grandes cantidades (incluso masivas) que perjudican de alguna o varias maneras al receptor.
SQLite	Sistema de gestión de bases de datos relacional
Symbian OS	Sistema operativo propiedad de Nokia
Tablet	Computadora portátil de mayor tamaño que un teléfono inteligente o un PDA, integrada en una pantalla táctil (sencilla o multitáctil) con la que se interactúa primariamente con los dedos o un estilete (pasivo o activo), sin necesidad de teclado físico ni ratón.
Windows Phone	Sistema operativo móvil desarrollado por Microsoft, como sucesor de Windows Mobile
Xcode	Entorno de desarrollo integrado de Apple Inc
YouTube	Sitio web en el cual los usuarios pueden subir y compartir vídeos

Tabla 1: Términos

1.6.2. ABREVIATURAS

ACRÓNIMO	DEFINICIÓN
UC3M	Universidad Carlos III de Madrid
PFC	Proyecto fin de carrera
TIC	Tecnologías de la información y la comunicación
XML	eXtensible Markup Language
RDF	Resource Description Framework (Marco de Descripción de Recursos)
PC	Personal Computer (Computadora Personal)
SDK	Software Development Kit (Kit de Desarrollo de Software)
API	Application Programming Interface (Interfaz de Programación de Aplicaciones)
MVC	Modelo Vista Controlador
IEEE	Institute of Electrical and Electronics Engineers (Instituto de Ingenieros Eléctricos y Electrónicos)
ERS	Especificación de Requisitos Software
URL	Uniform Resource Locator (Localizador Universal de Recursos)
UML	Unified Modeling Language (Lenguaje Unificado de Modelado)
RSS	Really Simple Syndication
OS	Operating System (Sistema Operativo)

Tabla 2: Abreviaturas

1.7. ESTRUCTURA DE LA MEMORIA

A continuación se explican los distintos capítulos que forman esta memoria y los contenidos principales en cada uno.

Capítulo 1: Introducción

Se basa en una breve introducción al objetivo de este Proyecto, describiendo las funcionalidades que presentan las comunicaciones RSS y la motivación para desarrollar una aplicación para implementar los servicios RSS. Aparte, se detallará la metodología utilizada y los medios necesarios para el desarrollo del Proyecto.

Capítulo 2: Estado del arte

Se aporta una visión del estado actual de los sistemas y tecnologías que utilizan servicios RSS, así como otras tecnologías similares existentes actualmente.

Asimismo se explicará la metodología empleada durante el proceso de desarrollo de la solución.

Capítulo 3: Especificación de requisitos

En este capítulo se detalla el alcance de la solución en base a los requisitos específicos de la aplicación. Se describirá con detalle cada requisito y su tipología.

Capítulo 4: Diseño de la solución

Descripción del diseño implementado para el desarrollo de la aplicación con tal de analizar la mejor solución posible, aportando casos de usos, diagramas de secuencias o modelos de datos.

Capítulo 5: Pruebas

Este capítulo detalla aquellas pruebas necesarias que se han ejecutado con tal de verificar la correcta funcionalidad de la aplicación.

Capítulo 6: Presupuesto

En este capítulo se expondrán los costes económicos y en tiempo asociados al desarrollo del proyecto.

Capítulo 7: Conclusiones

Este último capítulo muestra las conclusiones finales a las que se han llegado durante el transcurso del proyecto. También se expondrán las posibles evoluciones que pueden existir relativos al trabajo.

CAPÍTULO 2: ESTADO DEL ARTE

2.1. INTRODUCCIÓN A LOS LECTORES RSS

Hoy en día, existe una gran cantidad de sitios web o blogs que exponen su contenido a través de RSS feeds y permiten una “comunicación social” ágil y sencilla. Por esta razón, los lectores RSS se han convertido en un elemento imprescindible para que los usuarios puedan consultar su información favorita y tener centralizada todos los contenidos a los cuales se ha suscrito.

Un lector RSS (también llamado agregador) es el elemento para que los usuarios puedan recibir los feed de los sitios web o blogs sin tener que visitarlos, por lo que ordenan y gestionan el contenido de diversas fuentes de información y crear su propio “periódico virtual”.

Los lectores RSS simplemente transforman el contenido publicado en los medios que disponen de servicios RSS, en un formato de comunicación entendible por el lector. Existen varios tipos de lectores:

- Lectores web (a veces llamados portales).
- Lectores integrados en correo electrónico.
- Lectores integrados en los escritorios de los usuarios.
- Lectores de software dedicado.

Hay que destacar que existen lectores RSS con opciones de búsqueda de feeds. Esto permite facilitar la búsqueda de enlaces RSS de forma masiva, ya que muchos sitios web no publican explícitamente estos enlaces y requieren una búsqueda más concreta [3].

2.2. ¿QUÉ ES UN CANAL/SERVICIO RSS?

RSS significa Really Simple Syndication. RSS es un método eficaz para recibir actualizaciones (denominadas también fuentes RSS, fuentes de noticias o simplemente fuentes) de noticias y de información desde sus sitios web y blogs preferidos.

Las fuentes RSS consisten típicamente en titulares y pequeños resúmenes de nuevos artículos, nuevas entradas de un blog o resultados de búsqueda, si bien ciertos sitios web y blogs ofrecen el contenido íntegro de sus novedades [4].

Un canal RSS está construido con un subconjunto de instrucciones del lenguaje XML, un lenguaje de marcado extensible estricto de gran utilidad en el intercambio de datos, ya que permite describirlos sin mostrarlos al usuario, pero siendo a su vez legibles a través de diversas aplicaciones (navegadores, bases de datos, etc.)

2.3. ESTRUCTURA DE UN CANAL RSS

Cada mensaje RSS es un documento XML que tiene como nodo raíz un nodo <rss>. El nodo <rss> contiene un elemento <channel>. El elemento <channel> contendrá al menos los elementos <title>, <link> y <description>. Además el elemento channel podrá contener elementos <item>. Cada <item> es un artículo o noticia. La estructura general de un documento RSS es:

```
<rss version="2.0">
  <channel>
    <title>...</title>
    <link> ...</link>
    <description> ... </description>
    <item>
      <title>...</title>
      <link> ...</link>
      <description> ... </description>
    </item>
  </channel>
</rss>
```

2.3.1. ELEMENTOS DEL NODO CHANEL

Los elementos <title>, <link> y <description> son obligatorios. Además puede llevar elementos opcionales (language, copyright, managingEditor, webMaster,...)

2.3.2. ELEMENTOS DE LOS NODOS ITEM

Cada 'channel' puede llevar elementos 'item'. Cada 'item' informa de una noticia. En principio se trata de un título, una descripción y un enlace a la noticia completa. También puede pasar que el item tenga el contenido completo de la noticia en el elemento 'description'. En ese caso no es necesario rellenar los elementos 'title' y 'link'. Ningún elemento de 'item' es obligatorio, pero al menos uno de 'title', 'description' y 'link' tiene que llevar contenido.

```
<?xml version="1.0"?>
<rss version="2.0">
  <channel>
    <title>Sci-Fi Staff News</title>
    <link>http://liftoff.nasa.gov/</link>
    <description>Sci-Fi Staff News: Entertainment.</description>
    <language>en-us</language>
    <pubDate>Tue, 10 Jun 2003 04:00:00 GMT</pubDate>
    <lastBuildDate>Tue, 10 Jun 2003 04:41:01 GMT</lastBuildDate>
    <doc>http://blogs.law.harvard.edu/tech/rss</doc>
    <generator>Weblog Editor 2.5</generator>
    <managingEditor>editor@example.com</managingEditor>
    <webMaster>webmaster@example.com</webMaster>
    <item>
      <title>Solar Clive</title>
      <link>http://liftoff.nasa.gov/news/2003/news-solarclive.asp</link>
      <description>NASA's Astronauts get ready to work with Russians about the International Space Station? They look a little nervous in Russian, language and president of Russia?
      <clive href="http://www.ske.ssa.ru/STU/qtdo_english/Solar%20Clive/stu/</description>
      <pubDate>Tue, 03 Jun 2003 04:48:21 GMT</pubDate>
      <guid>http://liftoff.nasa.gov/2003/06/03.html#item73</guid>
    </item>
    <item>
      <description>Sky watchers in Europe, Asia, and parts of Alaska and Canada will experience a little
      href="http://headlines.nasa.gov/headlines/2003/03may_solarsclipse.htm">partial eclipse of the Sun?</a> on Saturday, May 31st.</description>
      <pubDate>Fri, 10 May 2003 11:06:42 GMT</pubDate>
      <guid>http://liftoff.nasa.gov/2003/05/10.html#item72</guid>
    </item>
    <title>The Engine That Does More</title>
    <link>http://liftoff.nasa.gov/news/2003/news_VASIMR.asp</link>
    <description>Mellor now travels to Mars, NASA hopes to design new engines that will let us fly through the Solar System more quickly. The proposed VASIMR engine
    would do the...</description>
    <pubDate>Tue, 27 May 2003 08:13:30 GMT</pubDate>
    <guid>http://liftoff.nasa.gov/2003/05/27.html#item71</guid>
    </item>
    <item>
      <title>Astronauts' Dirty Laundry</title>
      <link>http://liftoff.nasa.gov/news/2003/news-laundry.asp</link>
      <description>Compared to earlier spacecraft, the International Space Station has many luxuries, but laundry facilities are not one of them. Instead, astronauts have
      other options.</description>
      <pubDate>Tue, 20 May 2003 08:56:02 GMT</pubDate>
      <guid>http://liftoff.nasa.gov/2003/05/20.html#item70</guid>
    </item>
  </channel>
</rss>
```

Ilustración 1: Ejemplo XML fuente RSS 2.0

Además de estos tres elementos, el elemento 'item' también puede tener los siguientes elementos: author, category, comments, enclosure, guid, pubDate y source [5].

2.4. VERSIONES DE RSS

La historia de RSS ha sido agitada. Personas y grupos dispares han ido creado, prácticamente de modo independiente, especificaciones a las que llamaron RSS. Algunos formatos de RSS están basados en XML y otros lo están en RDF.

La lista mostrada a continuación muestra todas las versiones de RSS conocidas. Están listadas cronológicamente según fecha de publicación [6].

Nombre	Estado	Fecha de publicación	Basado en	Autor
RSS 0.90	Desaprobado	Marzo de 1999	RDF	Netscape
Netscape's RSS 0.91 (Revision 1)	Desaprobado	?	XML	Netscape
Netscape's RSS 0.91 Revision 3	Obsoleto	10 de Julio de 1999	XML	Netscape
Userland's RSS 0.91	Desaprobado	4 de Junio de 2000	XML	Userland
RSS 1.0	Estándar	9 de Diciembre de 2000	RDF	RSS-DEV Working Group
RSS 0.92	Desaprobado	25 de Diciembre de 2000	XML	Userland
RSS 0.93	Desaprobado	20 de Abril de 2001	XML	Userland
RSS 0.94	Desaprobado	19 de Agosto de 2002	XML	Userland
RSS 2.0	Desaprobado	Septiembre de 2002	XML	Userland
RSS 2.0 (post 2002-11-11)	Desaprobado	11 de Noviembre de 2002	XML	Userland
RSS 2.0 (post 2003-01-21)	Estándar	21 de Enero de 2003	XML	Userland

Ilustración 2: Versiones RSS

2.5. ¿CÓMO SABER SI UN SITIO WEB OFRECE SINDICACIÓN?

Si una página web ofrece también un canal RSS, se suele denotar con el siguiente icono:



Ilustración 3: Icono RSS

Curiosamente, los diseñadores de los dos navegadores más populares, Internet Explorer y Mozilla Firefox se han puesto de acuerdo para utilizar este icono.

2.6. ANÁLISIS DE APLICACIONES SIMILARES

En esta sección se procederá a hacer un análisis de aplicaciones existentes, desarrolladas con tecnologías similares, relacionadas con los objetivos del proyecto. Las aplicaciones analizadas, son de las más aceptación y usadas dentro del los sistemas Android & iOS. Se trata de un estudio muy básico del mercado, sobre las diferentes aplicaciones similares existentes durante la realización de este proyecto, con el único objetivo de valorar las semejanzas y diferencias entre las diferentes aplicaciones [7] [8].

2.6.1. FEEDLY

Feedly es una aplicación creada por DEvHD en 2008. Se trata de un lector RSS para navegadores webs y dispositivos móviles, que podemos encontrar tanto en Android como en iOS. Además, también ofrece un servicio basado en la nube.

Feedly se ha convertido en un lector muy autónomo al disponer de un contenido propio, lo que facilita la búsqueda de información. Por otra parte, se trata de un lector de uso gratuito que dispone de una enorme variedad de temas para personalizar la aplicación a las necesidades del usuario. Su apariencia es similar a Google Chrome, y con la desaparición de Google Reader se ha presentado como una aplicación de transición con una versión para Escritorio y extensión para Chrome.



Ilustración 4: Logo FEEDLY

2.6.2. FEEDDEMON

Se trata de un lector creado por NewsGator Technologies en 2005, que permite gestionar los enlaces RSS de manera centralizada y ordenada por el usuario. FeedDemon tiene carácter gratuito y posibilita que los usuarios puedan suscribirse a los feeds de su interés. Este lector dispone de una personalización propia para clasificar los feed por categorías y varias opciones como por ejemplo votar posts (entre Delicious o Diggs), distintos formatos de lectura, etc.

FeedDemon funciona para aplicaciones de Escritorio así como aplicaciones Web, con la misma configuración y el mismo sistema para sincronizar los contenidos con el lector RSS de NewsGator Online vía web. Esto permite que cualquier usuario pueda administrar los feed RSS mediante cualquier PC con acceso a Internet.



Ilustración 5: Logo FeedDemon

2.6.3. FLIPBOARD

Flipboard es una aplicación creada por Flipboard Inc. en el año 2010 de carácter gratuito.

Flipboard tiene una apariencia muy característica, que le hace destacar por ser muy visual y atractiva para el usuario. Este aspecto se trata de una ventaja comparándola con otros lectores disponibles en el mercado actual. Permite organizar los enlaces RSS como si se tratase de una revista digital, lo cual posibilita al usuario personalizar los contenidos preferidos y recibir las novedades de una manera muy sencilla.

Este lector RSS está disponible para la mayoría de sistemas operativos y dispositivos, como iOS, Windows Phone, Android, Blackberry, Kindle Fire y NOOK. Puede utilizarse tanto para Tablet, Smartphone o e-book, pero no dispone de versión web o escritorio, lo cual es una desventaja respecto a los otros dispositivos.

2.6.4. FUSION

Fusion es un lector RSS solo disponible para Windows Phone y de carácter "freemium", es decir, solo son gratuitas las funcionalidades básicas, mientras que las opciones avanzadas tienen un coste para aquellos usuarios que las precisen (son 1,29€).

Fusion permite además de gestionar y categorizar los feed RSS, unas funciones para integrarse con YouTube o soporte para vídeos. Además, posibilita almacenar los enlaces RSS en modo offline para su visualización en los momentos que no se disponga de conexión a Internet.



Ilustración 6: Logo FUSION

2.6.5. NETVIBES

Es una aplicación creada por Tariq Krim en 2005. Dispone de tres niveles de servicio con exclusividades distintas, para un perfil usuario, de empresa o de agencia. Solo el perfil usuario es de carácter gratuito. Netvibes ofrece una versión para escritorio y para móvil.

Respecto a sus funcionalidades, como lector RSS permite la gestión de los feed. No obstante, también ofrece la posibilidad de integrarse con redes sociales. Respecto a esta integración, dispone de un servicio de pago para visualizar las métricas de acceso a las redes sociales.



Ilustración 7: Logo NETVIBES

2.6.6. LINKEDIN PULSE

Se trata de un lector RSS desarrollado por dos estudiantes de Stanford y adquirida posteriormente por LinkedIn.

Funciona para dispositivos iPad y se caracteriza respecto a otros lectores RSS, por su apariencia particular que no expone los contenidos de manera tradicional. Ofrece un aspecto visual distinto que puede llegar a ser confuso para el propio usuario.

Es compatible para dispositivos iOS 7.0 o superior, tanto para iPhone, iPad e iPod Touch. Destacar su carácter gratuito en la App Store.



Ilustración 8: Logo LINKEDIN PULSE

2.7. ENTORNO DE TRABAJO SELECCIONADO

Según un estudio realizado por NetMarketShare [9], durante el mes de septiembre de 2015 los sistemas operativos móviles más utilizados, globalmente, fueron Android, iOS, y Windows Phone, entre otros.

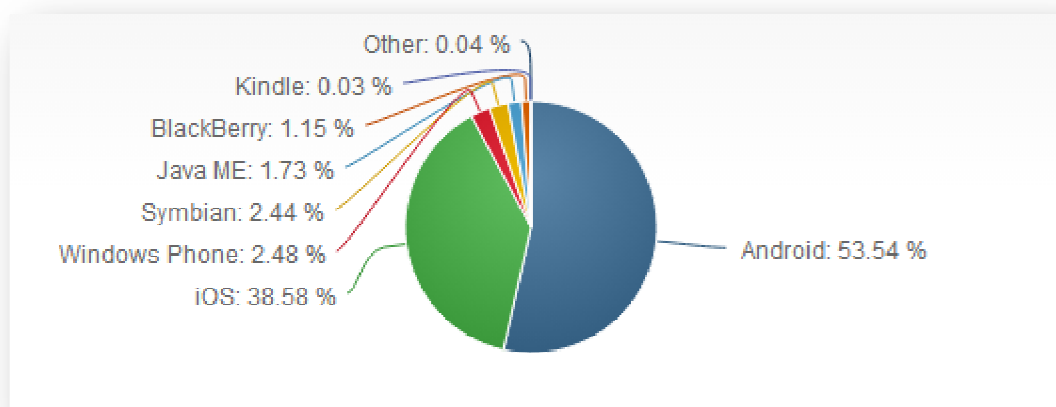


Ilustración 9: Sistemas operativos móviles más utilizados en 2015

Esta tendencia nos da una idea del enfoque que siguen los desarrollos para móviles, es decir, cuáles son los sistemas operativos más usados y, por tanto, a cuáles de estos sistemas operativos debe ir enfocado el desarrollo de aplicaciones móviles.

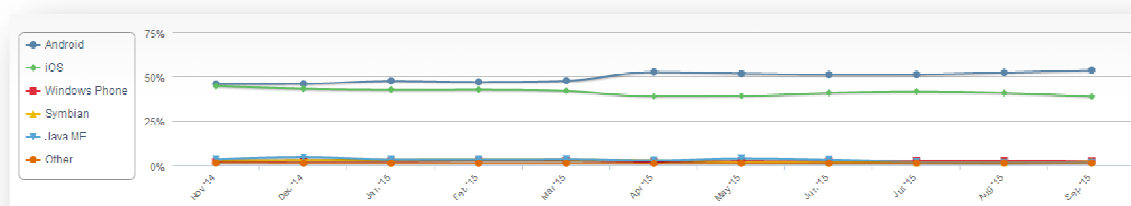


Ilustración 10: Sistemas operativos móviles más usados entre 2014 y 2015

Sin embargo, aunque la cuota de mercado de dispositivos Android es mayor, conviven multitud de versiones del sistema y tipos de dispositivos con multitudes de diferentes tamaños de pantalla y capacidades convirtiéndola en una plataforma para la es más difícil desarrollar.

Por el contrario, iOS proporciona una plataforma de desarrollo mucho menos fragmentada que acelera el desarrollo y despliegue de una aplicación. Aunque la cuota de mercado es menor que la de Android, está lo suficientemente extendida y consolidada como para lanzar una aplicación antes en esta plataforma, obtener datos significativos de uso que demuestren el éxito o fracaso de una idea, y según los resultados obtenidos decidir si merece la pena invertir los recursos necesarios en llevar la idea a Android para así poder abarcar el 92% del mercado de dispositivos móviles.



Ilustración 11: Logo APPLE & ANDROID

2.8. ENTORNO DE DESARROLLO IOS

iOS es el sistema operativo que emplean los dispositivos móviles de Apple, iPhone o iPad. El proceso de desarrollo para crear aplicaciones nativas para estos dispositivos se realiza a través de lo que se conoce como iOS Software Development Kit (SDK).

El conjunto de herramientas conocido como iOS Software Development Kit (SDK) contiene las herramientas e interfaces necesarias para desarrollar, instalar, ejecutar y probar aplicaciones nativas que aparecen en la pantalla de inicio de un dispositivo iOS. Las aplicaciones nativas son construidas empleando los frameworks del sistema iOS, el lenguaje de programación Objective-C y se ejecutan directamente sobre iOS.

2.8.1. ARQUITECTURA

La arquitectura iOS está basada en capas, las capas más altas contienen los servicios y tecnologías más importantes para el desarrollo de aplicaciones, y las capas más bajas controlan los servicios básicos.

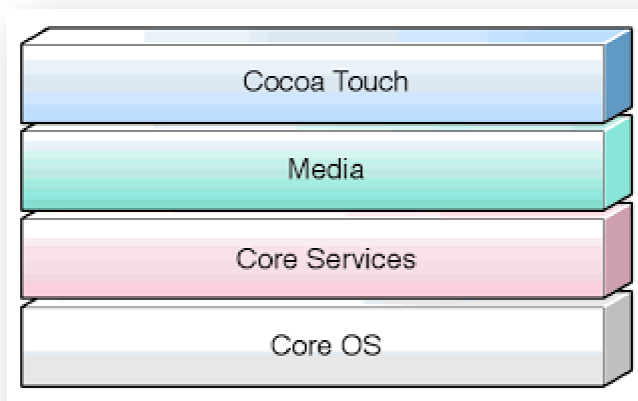


Ilustración 12: Arquitectura iOS

A continuación se expone brevemente los servicios más importantes que ofrece cada capa [10]:

- **Cocoa Touch:** Es la capa superior utilizada para interactuar con las aplicaciones, es decir, se trata de una capa visible que ofrece una capa de abstracción. Es la zona donde encontramos los componentes visuales. Cocoa Touch proviene de Cocoa, la API ya existente en la plataforma MAC y está formada por dos Frameworks fundamentales:
 - **UIKit:** contiene todas las clases que se necesitan para el desarrollo de una interfaz de usuario.
 - **Foundation Framework:** define las clases básicas, acceso y manejo de objetos, servicios del sistema operativo.
- **Media:** Se trata de una capa basada en la mezcla de lenguaje C y Objective C que contiene las tecnologías que dan acceso a ficheros multimedia relacionados con audio, gráficos, vídeos, etc.
- **Core Services:** Se trata de la capa de servicios principales disponibles en el dispositivo y que pueden ser utilizados por todas las aplicaciones, como pueden ser: base de datos SQLite, acceso a la red, soporte para XML.

- **Core OS:** El núcleo del sistema. Recordar que el sistema operativo iOS está basado en el OS X de Apple, que fue desarrollado a partir de una base Unix. Elementos de seguridad, memoria, procesos o manejo de ficheros son los que podemos encontrar en esta capa.

2.8.2. LENGUAJE DE PROGRAMACIÓN OBJECTIVE-C

Se trata de un lenguaje de programación desarrollado por la compañía Stepstonea principios de los 80, como una extensión del lenguaje C y otro lenguaje orientado a objetos, como Smalltalk.

A pesar de su antigüedad, no ha sido hasta muy recientemente cuando se ha popularizado su uso. Este crecimiento en los últimos años se debe a su adopción por parte de Apple que ha utilizado Objective-C como lenguaje de programación de su plataforma y frameworks de desarrollo. Esto ha obligado a los desarrolladores a utilizar Objective-C para poder implementar aplicaciones en iOS [11].

2.8.3. ENTORNO DE DESARROLLO X-CODE

Xcode se introdujo el 24 de octubre de 2003 junto con la versión 10.3 de Mac OS X, siendo desarrollado a partir del anterior entorno de desarrollo, Project Builder, al que sustituyó. Project Builder, a su vez, también era una herencia de la compañía NeXT, fusionada con Apple en 1996.



Ilustración 13: Logo X-CODE

Se trata de un entorno integrado de desarrollo moderno, con todas las funcionalidades que cabría esperar, tales como un editor de código fuente, autocompletado de código, un navegador de la estructura del proyecto, un compilador, un editor de interfaces gráficas, un simulador que permite reproducir tus desarrollos en varios tipos de dispositivos iPhone y iPad, entre muchas otras [12].

2.8.4. SQLITE

Sistema de bases de datos relacional, que permite realizar las operaciones básicas de un sistema gestor de base de datos (creación, consulta y actualización de tablas, relaciones y datos) y almacenarlas en un fichero de la aplicación. Es posible utilizarlo con otros sistemas como Android, Blackberry, Maemo, Symbian OS, etc.



Ilustración 14: Logo SQLITE

La principal diferencia con otros sistemas gestores de bases de datos es que SQLite no es un proceso independiente que sirve unos datos (servidor) y al que pueden conectarse las aplicaciones (clientes), sino que las bibliotecas SQLite se enlazan con la propia aplicación pasando a formar parte de ella.

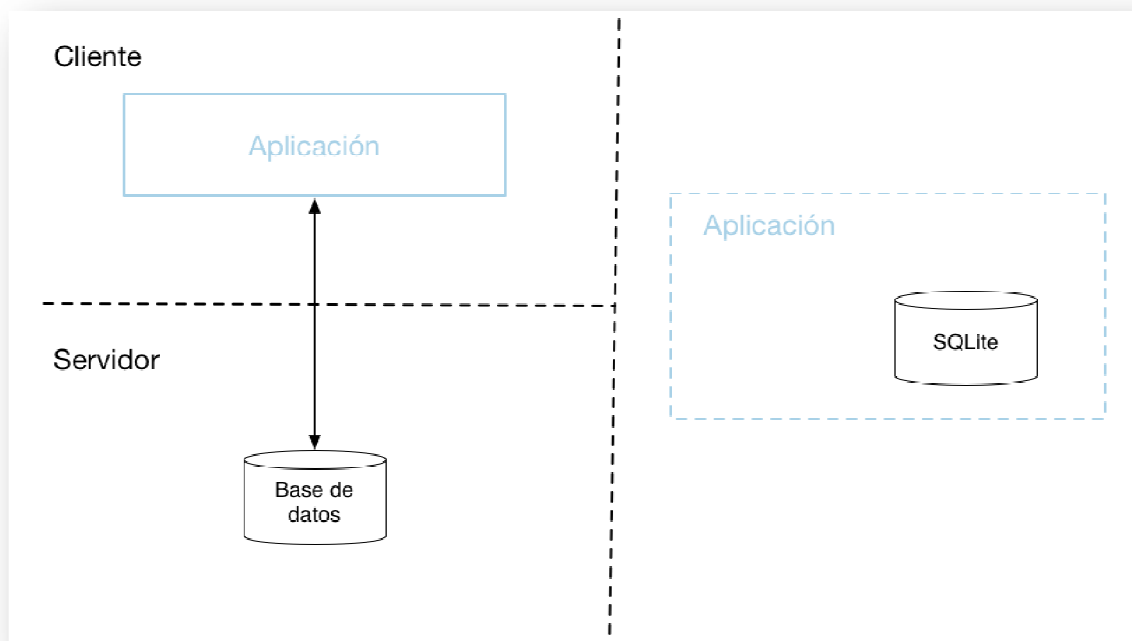


Ilustración 15: Diferencia entre un sistema gestor de base de datos tradicional y SQLITE

2.9. PATRONES DE DISEÑO UTILIZADOS

2.9.1. PATRÓN MODELO-VISTA-CONTROLADOR

Probablemente el patrón de diseño que más se conoce hoy día es el llamado patrón modelo-vista-controlador (o patrón MVC para simplificar de ahora en adelante). El nacimiento del mismo se remonta a 1979 en los laboratorios Xerox cuando se trabajaba con el lenguaje de programación orientado a objetos Smalltalk.

Se trata por tanto de un patrón que se usa en programación orientada a objetos en el que clasifica los mismos en uno de los siguientes roles: el modelo, la vista o el controlador.

Los objetos que se encuentran clasificados en el rol de vista, son aquellos que tienen que ver con la interfaz de usuario. Este tipo de objetos saben cómo presentarse en la pantalla, responden a las interacciones del usuario recogiendo sus acciones y mostrando los resultados de las mismas. Ejemplos de objetos que se encontrarían dentro del rol de vista podrían ser: botones, etiquetas o cuadros de texto.

Los objetos que cumplen con el rol del modelo son aquellos que van a contener los datos de la aplicación y definen la lógica y el proceso computacional que manipula y procesa los datos. Es decir, estos objetos van a encargarse de contener y manipular los datos de la aplicación. Ejemplos de objetos que se ajusten dentro de este rol serían: arrays que contengan los datos de la aplicación, o bien objetos que se encarguen de interactuar de forma directa con una base de datos.

Hasta ahora, los objetos definidos para la vista son los encargados de dibujar la interfaz de usuario, recoger las acciones de los usuarios y presentar los resultados. Por otra parte los objetos del modelo tratan con los datos almacenados de la aplicación, se encargan de su almacenamiento, su consulta y actualización. Sin embargo, hace falta un mecanismo capaz de poner en comunicación los objetos de la vista con los objetos del modelo. Es aquí donde surge la necesidad del controlador. Los objetos que encajan en esta categoría son los encargados de recoger las peticiones de los objetos de la vista y comunicárselas a los objetos del modelo, así como recoger los datos ofrecidos por los objetos del modelo y pasarlos a los objetos de la vista para que los presenten al usuario.

En resumen, los objetos de la vista recogen las peticiones del usuario y se las entregan a los objetos del controlador, los cuales saben cómo tienen que hacer la petición a los objetos del modelo. Estos por otra parte manejan los datos según la petición recibida y comunican el resultado a los objetos del controlador que son los encargados de pasar la respuesta a los objetos de la vista que, finalmente los presentan al usuario.

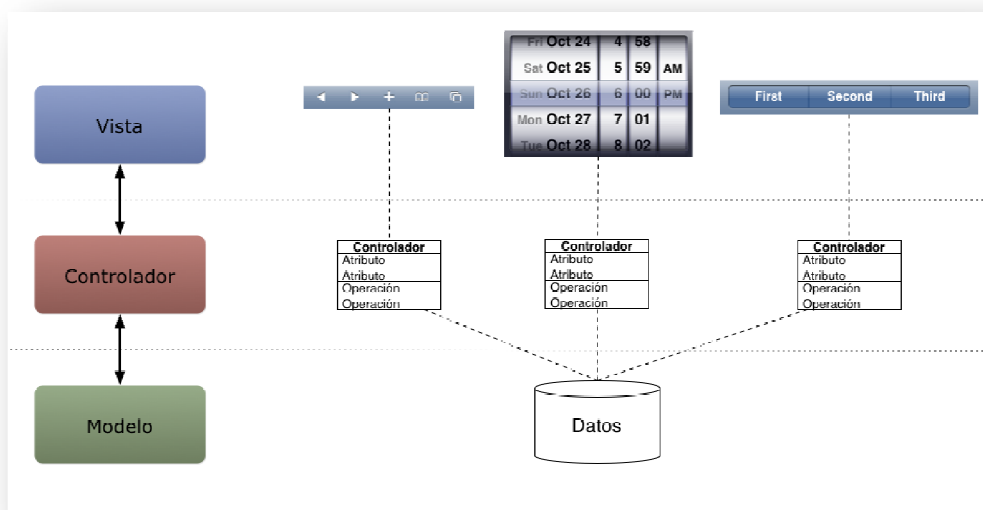


Ilustración 16: Patrón MODELO-VISTA-CONTROLADOR

Como puede observarse, este patrón de diseño no solo especifica el rol que va a cumplir cada objeto de la aplicación, sino que también determina la forma en la que van a comunicarse los mismos.

Este patrón de diseño es uno de los patrones más utilizados en el entorno de desarrollo de aplicaciones de Apple (Xcode) [13].

2.9.2. PATRÓN DELEGATE

En el patrón Delegate existe un objeto denominado delegate, en el cual confía otro objeto y al que va enviando las peticiones que necesite. Por tanto, en algún punto de la ejecución del programa, el primer objeto le manda al objeto en el que delega los mensajes solicitando cualquier tipo de información, y es este último objeto el que, implementado los métodos necesarios, contesta al primero con la información solicitada.

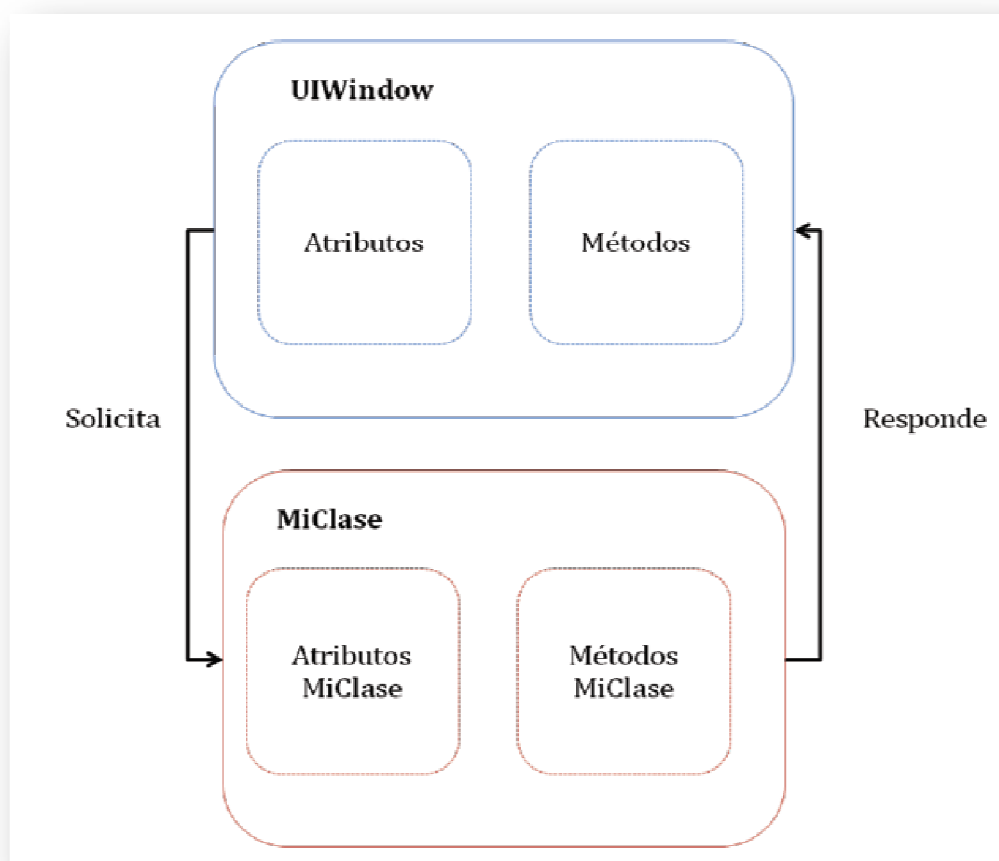


Ilustración 17: Delegate de funcionalidad para una ventana

El objeto que necesita un delegado para funcionar, debe marcar de alguna forma (generalmente en algún atributo), cuál es el objeto que va a actuar como delegado, teniendo así identificado a quién debe solicitar la información que necesite. Por otro lado el objeto que actúa como delegado, por ser precisamente delegado del primero, debe implementar una serie de métodos con un nombre y argumentos determinados. De esta forma, los objetos que actúan bajo este patrón de diseño saben perfectamente cómo comunicarse, ya que existe un protocolo definido entre ellos [14].

2.9.3. CENTRO DE NOTIFICACIONES

La comunicación entre los objetos que forman parte de una aplicación no tiene porque ser una comunicación uno a uno, es decir, que no siempre un objeto va a comunicarse con otro ejecutando un determinado método de su interfaz.

Imagínese por ejemplo una aplicación que recibe y gestiona correos electrónicos. En ella es evidente que va a existir un determinado objeto que se va a encargar de alguna manera de recibir los nuevos correos electrónicos que vayan entrando en el buzón. Pues bien, una vez que este objeto ha recibido el correo electrónico, debe avisar al resto de objetos de la aplicación de que hay un nuevo mensaje en la bandeja de entrada. Esta acción sin embargo puede hacerse de varias formas:

- La primera de ellas sería notificando a todos y cada uno de los objetos de la aplicación (o incluso del sistema) que hay un nuevo correo electrónico, de forma que cuando los objetos reciban este mensaje decidan si quieren actuar en consecuencia o por otro lado deciden ignorar el mensaje. El problema de esta aproximación, es que hay demasiado intercambio de mensajes en la aplicación ya que todos los objetos van a tener que informar de sus acciones al resto de objetos y van a ser estos últimos los que decidan qué hacer con dichos mensajes.

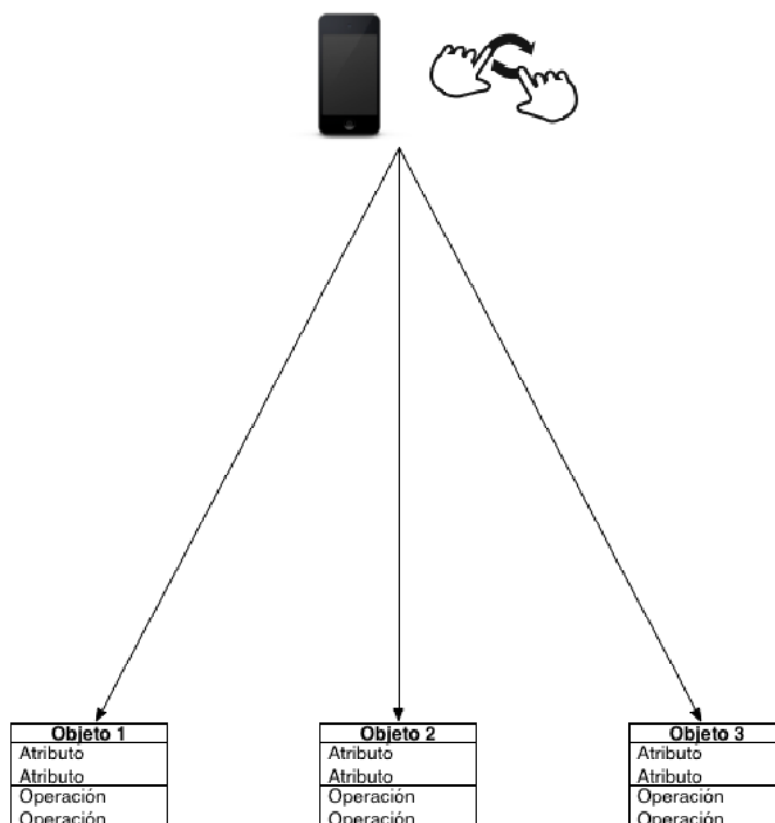


Ilustración 18: Aproximación sin centro de notificaciones

- Otra forma de solucionar el problema sería disponiendo de lo que se conoce como Centro de Notificaciones. Para comprender mejor lo que es este elemento imagínese una estación de trenes, en ella, cuando llega o sale un tren existe un tablón en el que se muestran las llegadas y las salidas al resto de personas de la estación. Es decir, no se va uno a uno de los viajeros que hay en la estación informándoles que existe un nuevo evento (una salida o llegada de un tren), sino que existe un elemento accesible por todos (tablón de anuncios) en que se van notificando todos esos eventos. Pues bien, el Centro de Notificaciones no es más que un "tablón" en el que se van registrando los eventos que suceden en el sistema (en realidad no se registran todos los eventos, solo aquellos que se desean registrar). De esta forma, los objetos interesados en estas notificaciones reciben todos al mismo tiempo la información [15].

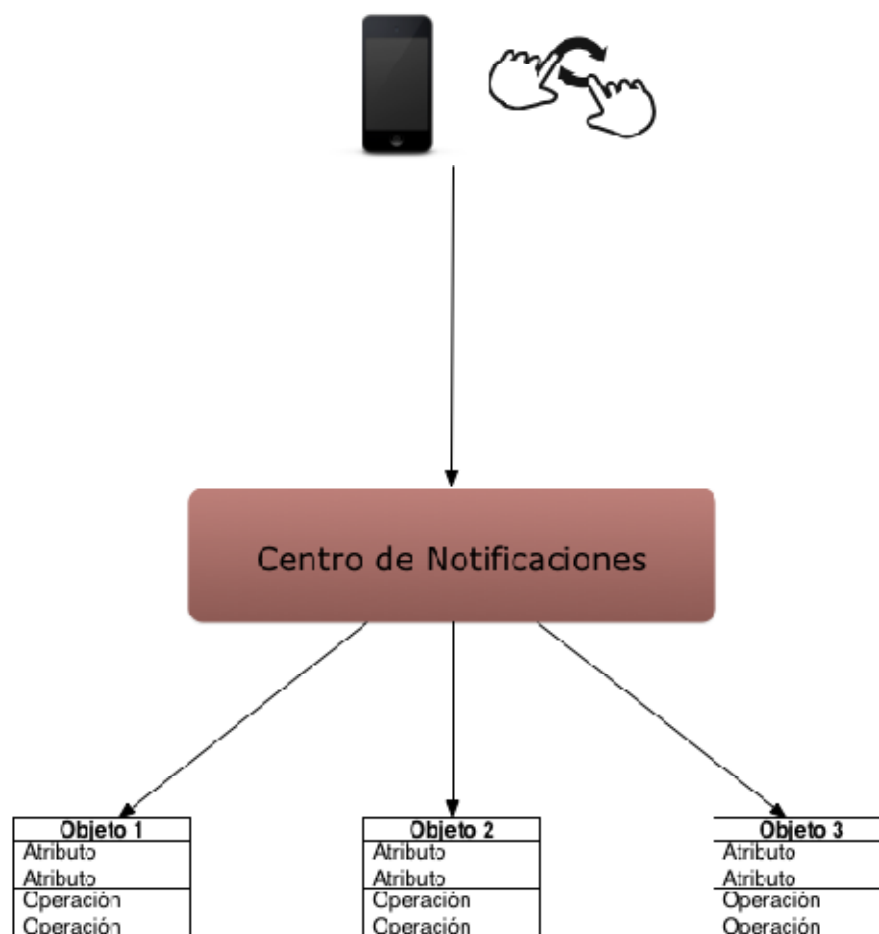


Ilustración 19: Uso del centro de notificaciones

2.9.4. TARGET-ACTION

Target-Action (o traducido como Objetivo-Acción) es muy sencillo de comprender. Cuando en la aplicación ocurre una determinada acción (por ejemplo, se pulsa un botón, se pulsa en un lugar de la pantalla, etc.), se crea un determinado mensaje utilizando para ello una serie de datos de interés y se envía a otro objeto. El mensaje que va a enviarse se conoce como nombre de selector y es la acción (Action) que va a llevarse a cabo. Por otro lado, el objeto que recibe la acción se denomina objetivo (Target).

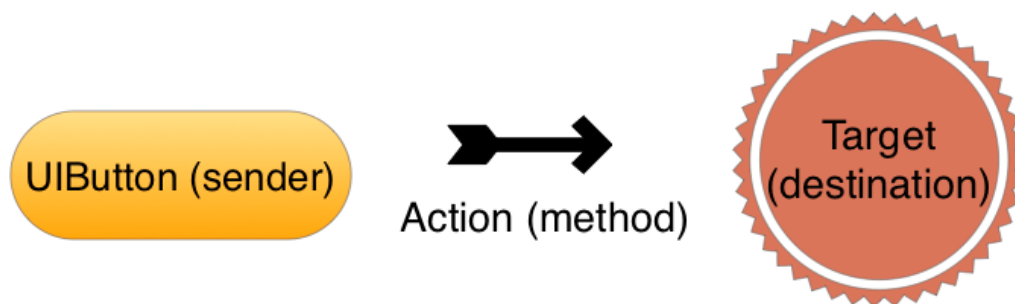


Ilustración 20: Patrón de diseño Target - Action

Es evidente la forma de funcionar de este patrón de diseño, aunque todavía se hace mucho más evidente si se piensa en la forma de funcionar de un smartphone actual, de forma que el usuario crea los eventos (mediante la interacción con el interfaz de usuario), y esos eventos son enviados a los objetos correspondientes de la aplicación para que los interpreten y den una respuesta adecuada [16].

CAPÍTULO 3: ESPECIFICACIÓN DE REQUISITOS

3.1. INTRODUCCIÓN

Este capítulo recoge la Especificación de Requisitos Software (ERS), una descripción completa del comportamiento del sistema. Para su elaboración se ha seguido el estándar IEEE 830.

3.2. DESCRIPCIÓN GENERAL

Siguiendo el estándar IEEE 830, en esta sección se especifica el alcance del sistema, cuya descripción servirá para hacer más fácil la comprensión de los aparados anteriores. Se busca definir el contexto en el que se sitúan los requisitos para poder definirlos con la mejor precisión.

3.2.1. PERSPECTIVA DEL PRODUCTO

Este producto se enmarca dentro del dominio de aplicaciones móviles y a su vez en lectores/agregadores RSS, aplicaciones que buscan reducir el tiempo y esfuerzo necesarios para comprobar/acceder a actualizaciones en las publicaciones de sitios web/blog, así como recopilar, centralizar y visualizar la información difundida a través de servicios RSS.

Esta aplicación es un producto independiente, no forma parte de ningún sistema mayor, pero interactúa con los servicios RSS publicados por infinidad de sitios web.

3.2.2. FUNCIONES DEL PRODUCTO

El producto solicitado presenta dos bloques de funcionalidad diferenciados:

- **Suscripción a servicios RSS:** los usuarios dispondrán de las herramientas necesarias para poder gestionar la conexión con los servicios RSS deseados de manera sencilla.
- **Visualización** de la información actualizada publicada por los servicios RSS suscritos previamente.

3.2.3. CARACTERÍSTICAS DE LOS USUARIOS

La aplicación se enfocará para cualquier usuario que disponga de un dispositivo iPad, con conocimientos básicos sobre su uso, tales como instalar aplicaciones o interacciones básicas como navegar por las distintas pantallas de una aplicación, pulsar botones, introducir texto..etc.

3.2.4. RESTRICCIONES

El producto será desarrollado para plataforma iOS, por lo que se presentan las siguientes restricciones:

- El producto debe ser desarrollado empleando el kit de desarrollo proporcionado por Apple, que únicamente funciona en equipos con sistema operativo Mac OS X.
- El sistema operativo Mac X está soportado únicamente por ordenadores Apple.
- El lenguaje de programación a emplear debe ser Objective-C, único lenguaje soportado oficialmente por el kit de desarrollo.
- El producto debe seguir la guía de estilo para interfaces de usuario publicadas por Apple.

- El producto únicamente aceptará publicaciones emitidas por servicios RSS 2.0.

3.2.5. SUPOSICIONES Y DEPENDENCIAS

La aplicación se podrá ejecutar en cualquier dispositivo iPad, siendo necesaria conexión a internet para que el producto responda correctamente.

Además, la aplicación presenta una dependencia sumamente importante del estado de los servidores en los que se alojan los servicios RSS suscritos. Si dichos servicios no están disponibles, la aplicación no podrá visualizar ningún tipo de información actualizada del servicio en cuestión.

3.2.6. ESPECIFICACIÓN DE REQUISITOS

A continuación se describen todos los requisitos que forman parte del proyecto para consolidar una especificación completa a partir de la cual diseñar la solución al problema que nos atañe.

Para la especificación de requisitos expuestos en este documento, se utilizará una tabla como la expuesta a continuación.

Identificador	
Nombre	
Descripción	
Prioridad	

Tabla 3: Ejemplo para especificación de requisitos

- **Identificador:** código que identifica de manera única al requisito. El formato establecido para identificarlos seguirá el siguiente patrón:
 - RF-XX: requisitos funcionales.
 - RNF-XX: requisitos no funcionales.
 - RD-XX: requisitos de diseño.

Donde '**XX**' se sustituirá por numeración ascendente, empezando por 01.

- **Nombre:** nombre identificativo del requisito.
- **Descripción:** pequeña reseña acerca del requisito.

3.2.6.1. REQUISITOS FUNCIONALES

En este apartado se describen todas las funcionalidades que debe contener la solución.

Identificador	RF-01
Nombre	Suscribirse a un RSS
Descripción	<p>Los usuarios podrán suscribirse a feeds de sus sitios web favoritos. Se podrá agregar un RSS a partir de los siguientes datos:</p> <p>Nombre</p> <p>URL seleccionada previamente.</p>

Tabla 4: RF-01 Suscribirse a un RSS

Identificador	RF-02
Nombre	Borrar RSS
Descripción	Los usuarios podrán borrar feeds a los que se han suscrito previamente.

Tabla 5: RF-02 Borrar RSS

Identificador	RF-03
Nombre	Validar RSS
Descripción	La aplicación implementada deberá ser capaz de identificar que los RSS a los que se suscriben los usuarios pertenecen a la versión 2.0

Tabla 6: RF-03 Validar RSS

Identificador	RF-04
Nombre	Listar RSS suscritos
Descripción	Los usuarios deben poder consultar los RSS a los que están suscritos.

Tabla 7: RF-04 Listar RSS suscritos

Identificador	RF-05
Nombre	Listar feeds por RSS
Descripción	Los usuarios deben poder consultar el listado de feeds de un RSS al que se ha suscrito previamente.

Tabla 8: RF-05 Listar feeds por RSS

Identificador	RF-06
Nombre	Visualizar feed.
Descripción	Los usuarios podrán visualizar de manera individualizada cada una de los feeds asociados a un RSS.

Tabla 9: RF-06 Visualizar feed

Identificador	RF-07
Nombre	Visualizar feed en sitio web.
Descripción	Los usuarios podrán visualizar cada uno de los feed dentro del sitio web que lo publica.

Tabla 10: RF-07 Visualizar feed en sitio web

Identificador	RF-08
Nombre	Archivar feed.
Descripción	Los usuarios podrán archivar sus feeds favoritos.

Tabla 11: RF-08 Archivar feed

Identificador	RF-09
Nombre	Actualizar feeds.
Descripción	Los usuarios podrán actualizar cuando deseen los feeds de los RSS a los que está suscrito.

Tabla 12: RF-09 Actualizar feeds

Identificador	RF-10
Nombre	Listar feeds archivados.
Descripción	Los usuarios deben poder consultar el listado de los feeds archivados previamente.

Tabla 13: RF-10 Listar feeds archivados

Identificador	RF-11
Nombre	Marcar feed como no leído.
Descripción	Los usuarios deben poder marcar los feeds archivados, cómo no leídos y por tanto desaparecer del listado de archivados.

Tabla 14: RF-11 Marcar feed como no leído

3.2.6.2. REQUISITOS NO FUNCIONALES

A continuación se exponen los requisitos que imponen restricciones en el diseño o la implementación más allá de las funcionalidades expuestas.

Identificador	RNF-01
Nombre	Dispositivos soportados.
Descripción	La solución debe funcionar con dispositivos que funcionen con el sistema operativo IOS, concretamente iPADS.

Tabla 15: RNF-01 Dispositivos soportados

Identificador	RNF-02
Nombre	Versión mínima del sistema operativo IOS.
Descripción	La versión mínima del sistema IOS sobre la que funcionará la solución es la versión XX.

Tabla 16: RNF-02 Versión mínima soportada

Identificador	RNF-03
Nombre	Uso del patrón de interfaz Pull to refresh
Descripción	La actualización de feeds deberá solicitarse manualmente empleando el patrón Pull to refresh

Tabla 17: RNF-03 Uso patrón Pull to Refresh

Identificador	RNF-04
Nombre	Aplicación accesible a través de internet.
Descripción	La aplicación necesitará de conexión a internet para funcionar correctamente.

Tabla 18: RNF-04 Accesible a través de internet

Identificador	RNF-05
Nombre	Cambios de orientación.
Descripción	La aplicación deberá permitir cambios de orientación: vertical → horizontal y viceversa.

Tabla 19: RNF-05 Cambio de orientación

3.3. ESPECIFICACIÓN DE CASOS DE USO

A continuación se describen los casos de uso identificados en el proyecto, cada uno de ellos proporciona un escenario que indica cómo debería interactuar el sistema con el usuario.

Inicialmente se presentará el diagrama de casos de uso en lenguaje gráfico UML y a continuación se procederá a realizar una descripción textual de cada caso de uso.

Para la descripción textual se empleará una tabla con los siguientes campos:

- **Identificador:** identifica de forma única cada caso de uso. El formato utilizado es el siguiente "CU-XX", donde 'XX' se sustituirá por numeración ascendente, empezando por 01.
- **Nombre:** nombre asignado al caso de uso.
- **Actores:** representa los distintos roles de usuarios que encontraremos en el sistema. Cualquier entidad externa al sistema que está relacionado con éste y que le demanda una funcionalidad. En este caso no existen diferenciaciones de roles, todos los usuarios serán tipificados con el mismo perfil, que identificaremos como "lector".
- **Objetivo:** finalidad que se pretende alcanzar tras ejecutarse el caso de uso.
- **Pre-condición:** condiciones iniciales que se deben cumplir para que se ejecute el caso de uso.

- **Post-condición:** condiciones finales que se cumplirán tras la ejecución del caso de uso.
- **Escenario básico:** secuencia de pasos que hay que seguir, de forma ordenada, en un flujo normal del caso de uso.
- **Escenario alternativo:** secuencias de pasos que pueden realizarse en lugar del funcionamiento normal. Generalmente hacen referencia a caminos incorrectos o errores.

3.3.1. DIAGRAMA DE CASOS DE USO

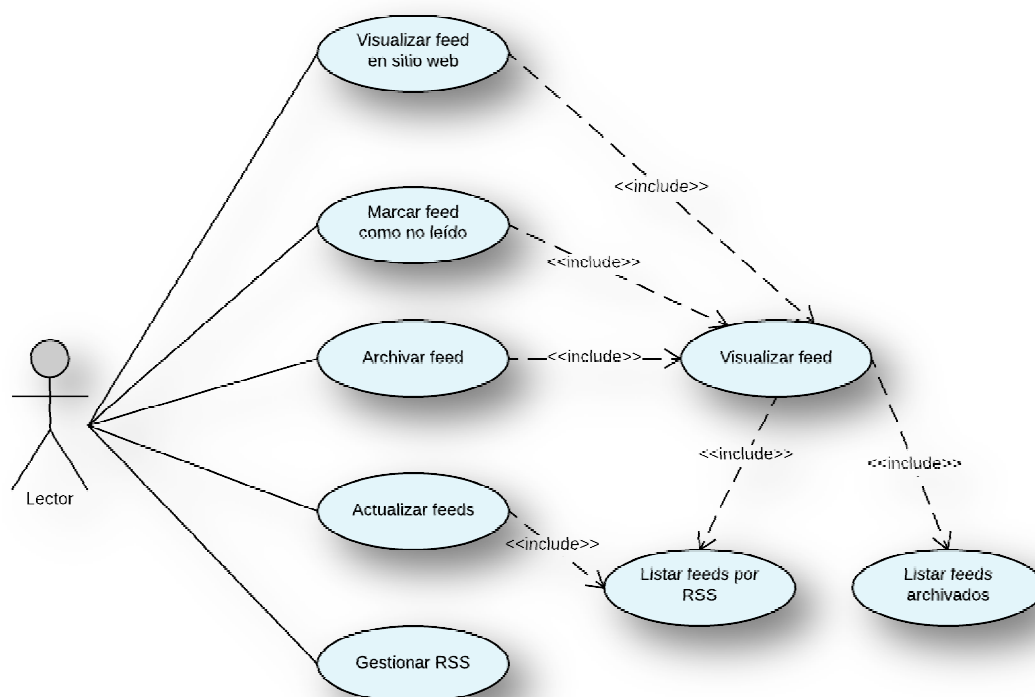


Ilustración 21: Diagrama de casos de uso

3.3.2. DESCRIPCIÓN CASOS DE USO

Identificador	CU-01
Nombre	Gestionar RSS
Actores	Lector
Objetivo	El lector necesita dar de alta, baja y consultar un enlace a un servicio RSS.
Pre-condición	La aplicación debe encontrarse iniciada. El menú principal debe estar visible.
Post-condición	La aplicación registra los cambios realizados.
Escenario básico	Acceder al menú principal. Mostrar la pantalla de gestión de RSS. Insertar datos obligatorios. Realizar acción deseada.
Escenario alternativo	4a. No se rellenan los campos obligatorios. 4b. La conexión con la base de datos es incorrecta.

Tabla 20: CU-01 Gestionar RSS

Identificador	CU-02
Nombre	Listar feeds por RSS
Actores	Lector
Objetivo	El lector necesita listar los feeds asociados a los enlaces RSS suscritos previamente.
Pre-condición	La aplicación debe encontrarse iniciada y conectada a internet. El menú principal debe estar visible. Haber dado de alta un enlace a un servicio RSS.

Post-condición	La aplicación muestra una lista de feeds publicados por los servicios RSS suscritos.
Escenario básico	Acceder al menú principal. Mostrar el listado de feeds por RSS
Escenario alternativo	2a. La conexión con la base de datos es incorrecta. 2b.No se puede conectar con el servidor de los enlaces RSS suscritos.

Tabla 21: CU-02 Listar feeds por RSS

Identificador	CU-03
Nombre	Visualizar feed
Actores	Lector
Objetivo	El lector necesita visualizar la información publicada en un feed.
Pre-condición	La aplicación debe encontrarse iniciada y conectada a internet. El menú principal debe estar visible. Haber dado de alta un enlace RSS
Post-condición	La aplicación muestra la información del feed seleccionado.
Escenario básico	Acceder al menú principal. Mostrar el listado de feeds por RSS Seleccionar un feed del listado.
Escenario alternativo	2a. La conexión con la base de datos es incorrecta. 2b.No se puede conectar con el servidor de los enlaces RSS suscritos.

Tabla 22: CU-03 Visualizar feed

Identificador	CU-04
Nombre	Actualizar feeds
Actores	Lector
Objetivo	El lector necesita actualizar los feeds de los enlaces a servicios RSS suscritos, para recuperar los feeds actualizados publicados por dichos servicios.
Pre-condición	La aplicación debe encontrarse iniciada y conectada a internet. El menú principal debe estar visible. Haber dado de alta un enlace RSS
Post-condición	La aplicación actualiza la información de los enlaces RSS suscritos y muestra el listado de feeds por RSS actualizado.
Escenario básico	Acceder al menú principal. Mostrar el listado de feeds por RSS Actualizar feeds
Escenario alternativo	2a. No se puede conectar con el servidor de los enlaces RSS suscritos. 3a. No se puede conectar con el servidor de los enlaces RSS suscritos

Tabla 23: CU-04 Actualizar feeds

Identificador	CU-05
Nombre	Listar feeds archivados
Actores	Lector
Objetivo	El lector necesita listar los feeds archivados previamente.

Pre-condición	<p>La aplicación debe encontrarse iniciada.</p> <p>El menú principal debe estar visible.</p> <p>Haber dado de alta un enlace RSS.</p> <p>Haber archivado un feed</p>
Post-condición	La aplicación muestra un listado de feeds archivados
Escenario básico	<p>Acceder al menú principal.</p> <p>Mostrar el listado de feeds archivados</p>
Escenario alternativo	2a. La conexión con la base es incorrecta.

Tabla 24: CU-05 Listar feeds archivados

Identificador	CU-06
Nombre	Marcar feed como no leído
Actores	Lector
Objetivo	El lector necesita eliminar un feed del listado de feeds archivados.
Pre-condición	<p>La aplicación debe encontrarse iniciada.</p> <p>El menú principal debe estar visible.</p> <p>Haber dado de alta un enlace RSS.</p> <p>Haber archivado un feed</p>
Post-condición	La aplicación registra los cambios realizados.
Escenario básico	<p>Acceder al menú principal.</p> <p>Mostrar el listado de feeds archivados.</p> <p>Seleccionar feed del listado.</p> <p>Visualizar feed.</p> <p>Borrar el feed seleccionado.</p>
Escenario alternativo	<p>2a. La conexión con la base de datos es incorrecta</p> <p>5a. La conexión con la base de datos no es correcta.</p>

Tabla 25: CU-06 Marcar feed como no leído

Identificador	CU-07
Nombre	Archivar feed
Actores	Lector
Objetivo	El lector necesita archivar un feed leído
Pre-condición	La aplicación debe encontrarse iniciada. El menú principal debe estar visible. Haber dado de alta un enlace RSS. Haber seleccionado un feed para visualizar
Post-condición	La aplicación registra los cambios realizados.
Escenario básico	Acceder al menú principal. Mostrar el listado de feeds por RSS. Seleccionar un feed del listado. Visualizar feed Archivar feed
Escenario alternativo	2a. No se puede conectar con el servidor de los enlaces RSS suscritos. 5a. La conexión con la base de datos es incorrecta

Tabla 26: CU-07 Archivar feed

Identificador	CU-08
Nombre	Visualizar feed en sitio web
Actores	Lector
Objetivo	El lector necesita poder abrir un feed seleccionado en el sitio web de su publicación
Pre-condición	La aplicación debe encontrarse iniciada. El menú principal debe estar visible.

	<p>Haber dado de alta un enlace RSS.</p> <p>Haber seleccionado un feed para visualizar</p>
Post-condición	La aplicación muestra el feed en el sitio web que lo publica.
Escenario básico	<p>Acceder al menú principal.</p> <p>Mostrar el listado de feeds por RSS.</p> <p>Seleccionar un feed del listado.</p> <p>Visualizar feed</p> <p>Ver feed en sitio web</p>
Escenario alternativo	<p>2a. La conexión con la base de datos es incorrecta</p> <p>2b. No se puede conectar con el servidor de los enlaces RSS suscritos.</p> <p>5a. No se puede conectar con el sitio web que publica el feed seleccionado.</p>

Tabla 27: CU-08 Visualizar feed en sitio web

3.3.3. TRAZABILIDAD REQUISITOS-CASOS DE USO

	CU-01	CU-02	CU-03	CU-04	CU-05	CU-06	CU-07	CU-08
RF-01	X							
RF-02	X							
RF-03	X							
RF-04	X							
RF-05		X						
RF-06			X					
RF-07								X
RF-08							X	
RF-09				X				
RF-10					X			
RF-11						X		

Tabla 28: Trazabilidad requisitos-casos de uso

CAPITULO 4: DISEÑO DE LA SOLUCIÓN

4.1. INTRODUCCIÓN

En el presente capítulo se van a mostrar todo los diagramas UML necesarios para poder llevar a cabo el correcto desarrollo de la aplicación, para ello es necesario basarse en todos los aspectos resultantes del anterior análisis de la aplicación.

4.2. DIAGRAMAS DE SECUENCIA

A continuación se muestran las interacciones entre los distintos componentes de la aplicación.

Los diagramas de han estructurado siguiendo la especificación de casos de uso, es decir, cada diagrama de secuencia engloba la funcionalidad en cada caso de uso.

4.2.1. GESTIONAR RSS

Es este apartado se representa la secuencia de pasos que sigue un usuario para gestionar sus enlaces a servicios RSS. Para ello accederá al menú principal, seleccionará la opción RSS, que mostrará la pantalla gestión RSS, dónde será posible suscribirse, borrar o consultas los enlaces RSS.

4.2.1.1. ALTA ENLACE RSS

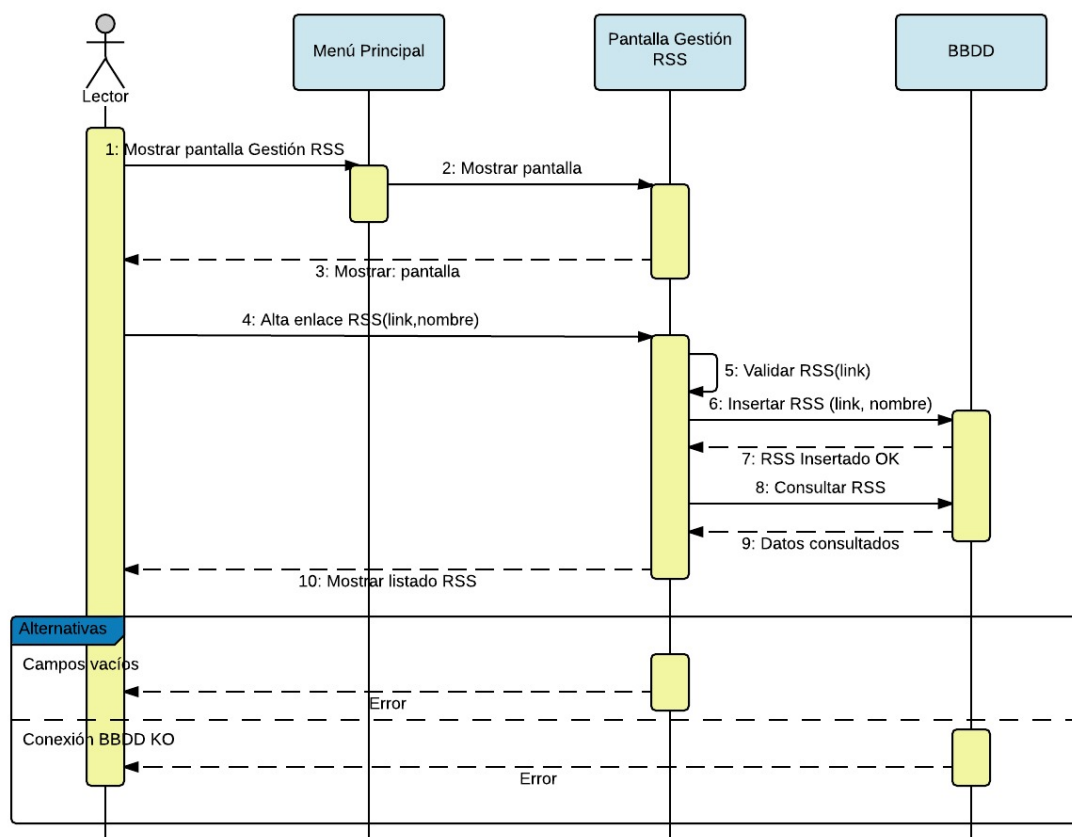


Ilustración 22: Diagrama de secuencia alta enlace RSS

4.2.1.2. BORRAR ENLACE RSS

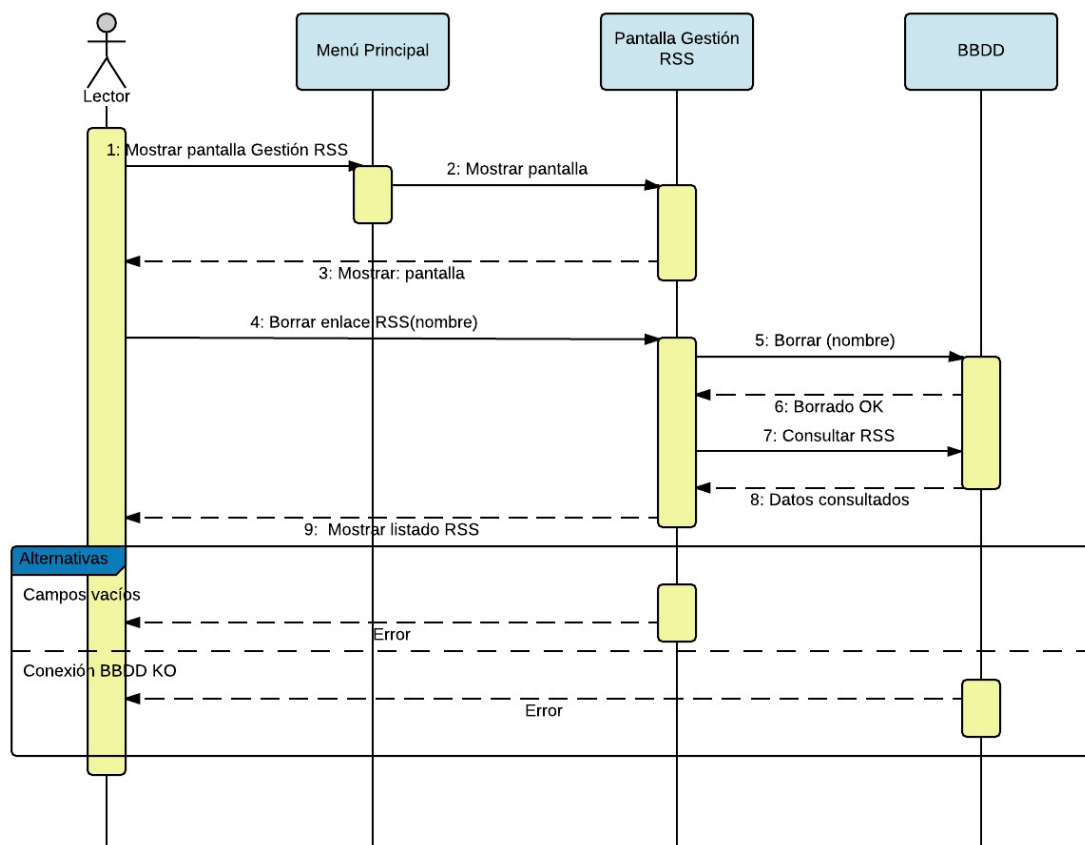


Ilustración 23: Diagrama de secuencia baja enlace RSS

4.2.1.3. MOSTRAR ENLACE RSS

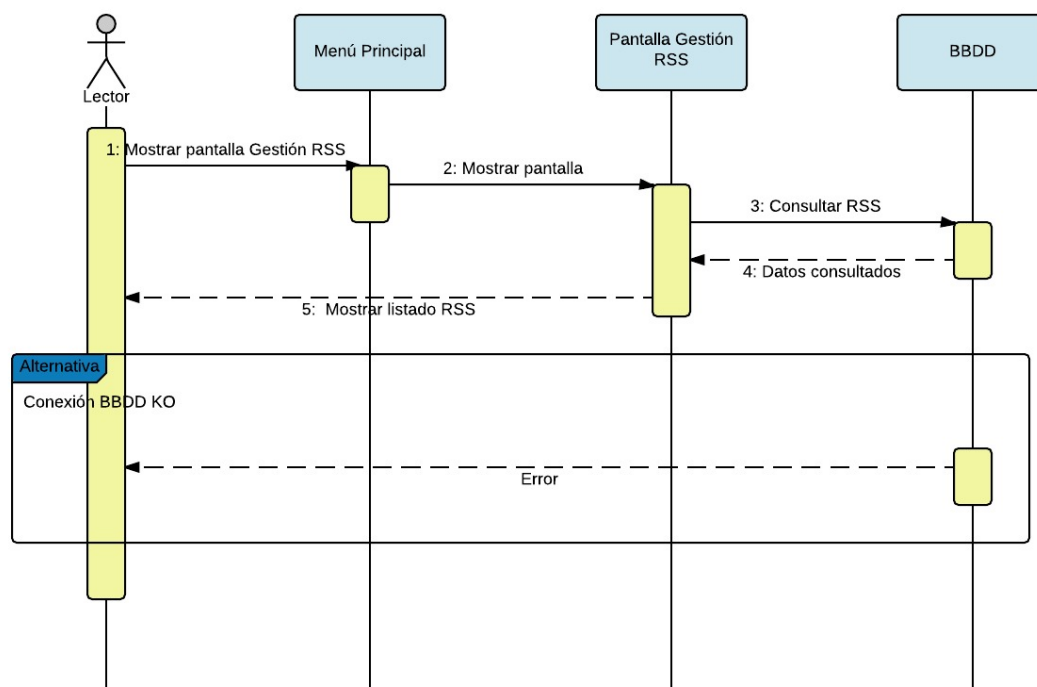


Ilustración 24: diagrama de secuencia mostrar enlace RSS

4.2.2. LISTAR FEEDS POR RSS

Es este apartado se representa la secuencia de pasos que sigue un usuario para listar los feeds asociados a los enlaces RSS suscritos previamente. Para ello accederá al menú principal, seleccionará la opción FEEDS que mostrará la pantalla con el listado de feeds.

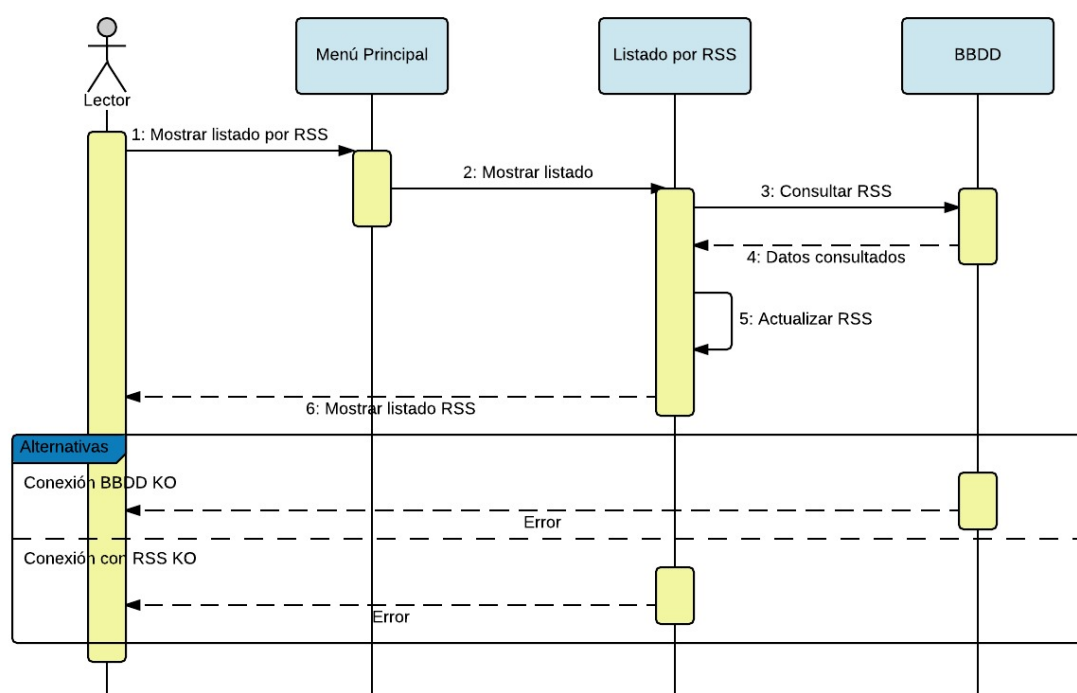


Ilustración 25: Diagrama de secuencia listar feeds por RSS

4.2.3. VISUALIZAR FEED

Es este apartado se representa la secuencia de pasos que sigue un usuario para visualizar la información de un feed. Para ello accederá al menú principal, seleccionará la opción FEEDS, que mostrará la pantalla con el listado de feeds disponibles, finalmente se seleccionará del listado, el feed que interese visualizar.

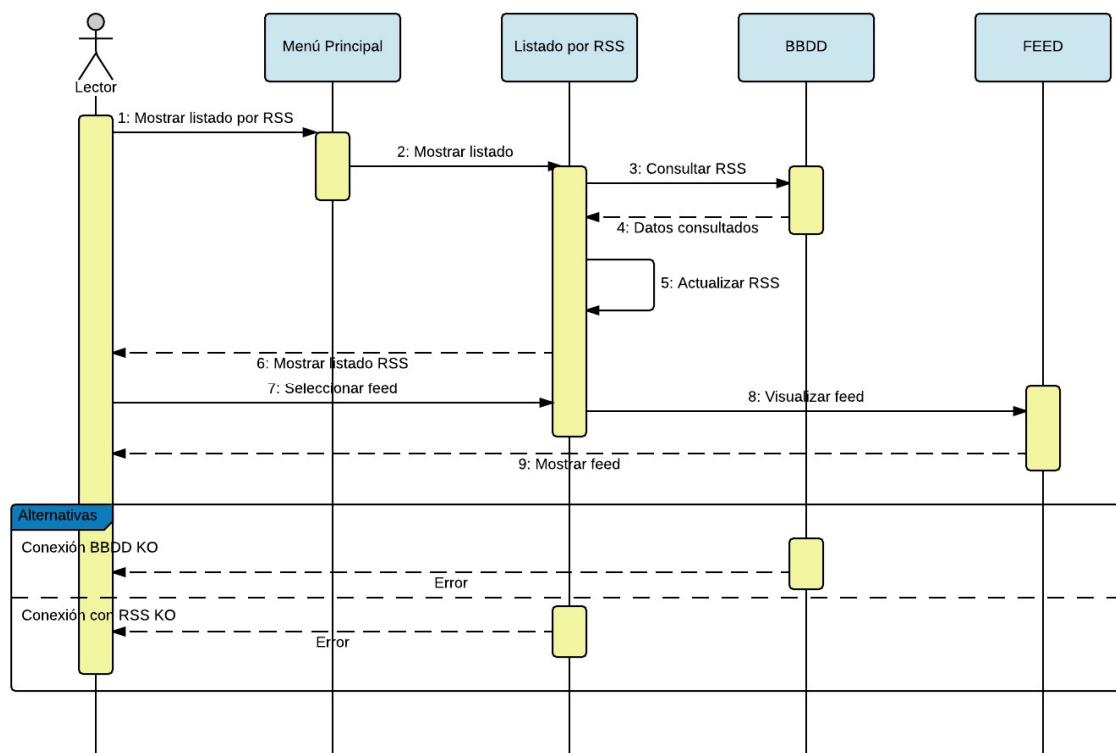


Ilustración 26: Diagrama de secuencia visualizar feed

4.2.4. ACTUALIZAR FEEDS

Es este apartado se representa la secuencia de pasos que sigue un usuario para actualizar los feeds publicados por un enlace RSS suscrito anteriormente. Para ello accederá al menú principal, seleccionará la opción FEEDS, que mostrará la pantalla con el listado de feeds disponibles, mediante el patrón Pull to refresh se procederá a la actualización de feeds.

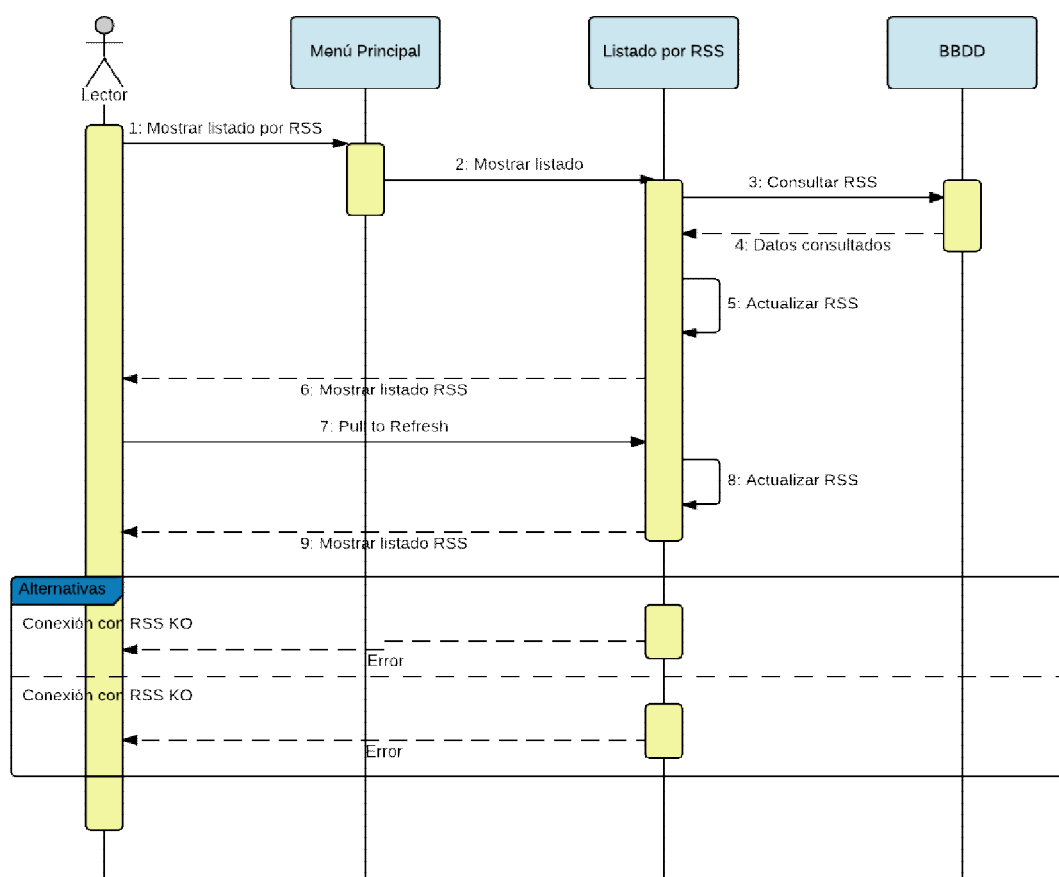


Ilustración 27: Diagrama de secuencia actualizar feeds

4.2.5. LISTAR FEEDS ARCHIVADOS

Es este apartado se representa la secuencia de pasos que sigue un usuario para listar los feeds archivados. Para ello accederá al menú principal, seleccionará la opción ARCHIVO, que mostrará la pantalla con el listado de feeds archivados.

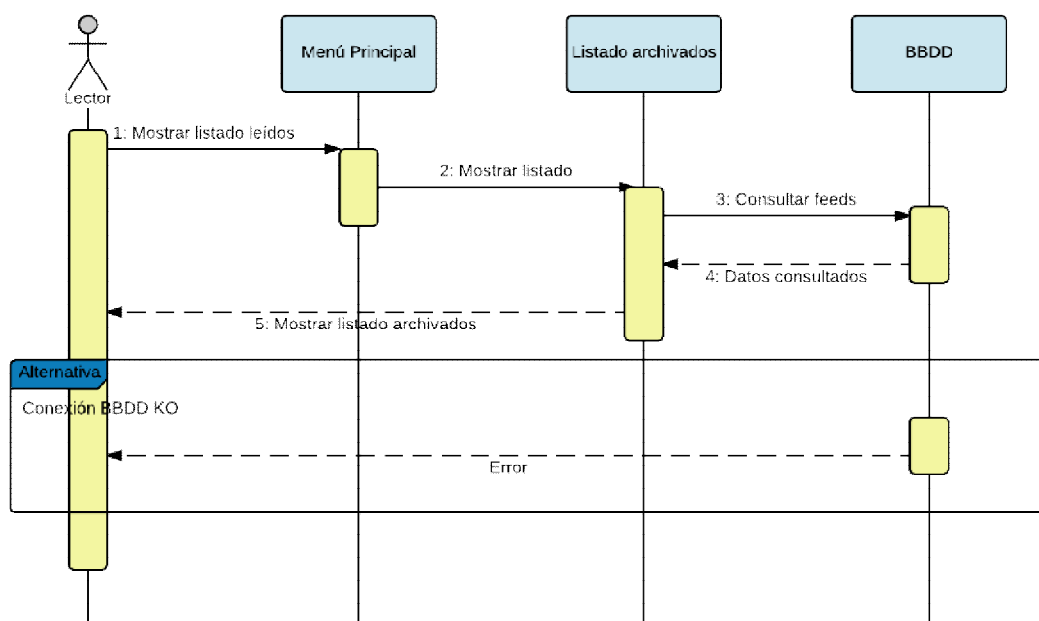


Ilustración 28: Diagrama de secuencia listar feeds archivados

4.2.6. MARCAR FEED COMO NO LEÍDO

Es este apartado se representa la secuencia de pasos que sigue un usuario para marcar un feed del archivo, como no leído y como consecuencia, desaparezca de dicho listado. Para ello accederá al menú principal, seleccionará la opción ARCHIVO, que mostrará la pantalla con el listado de feeds archivados, finalmente se seleccionará el feed que desee marcar y borrar del listado.

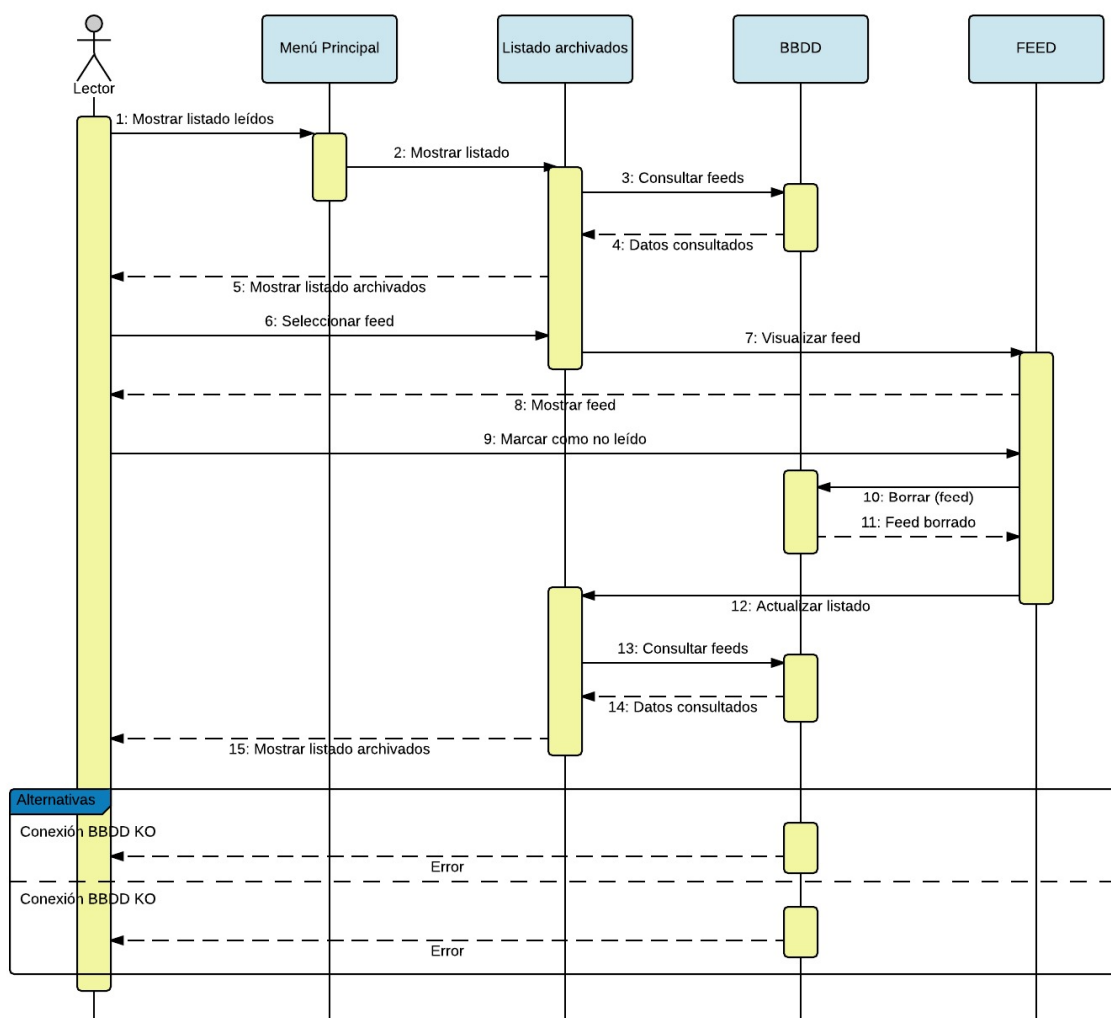


Ilustración 29: Diagrama de secuencia borrar feed archivado

4.2.7. ARCHIVAR FEED

Es este apartado se representa la secuencia de pasos que sigue un usuario para archivar un feed que se está visualizando, y como consecuencia que aparezca en el listado de feeds archivados. Para ello accederá al menú principal, seleccionará la opción FEEDS, que mostrará la pantalla con el listado de feeds disponibles, finalmente se seleccionará del listado, el feed que interese visualizar y seleccionará la opción de archivar feed.

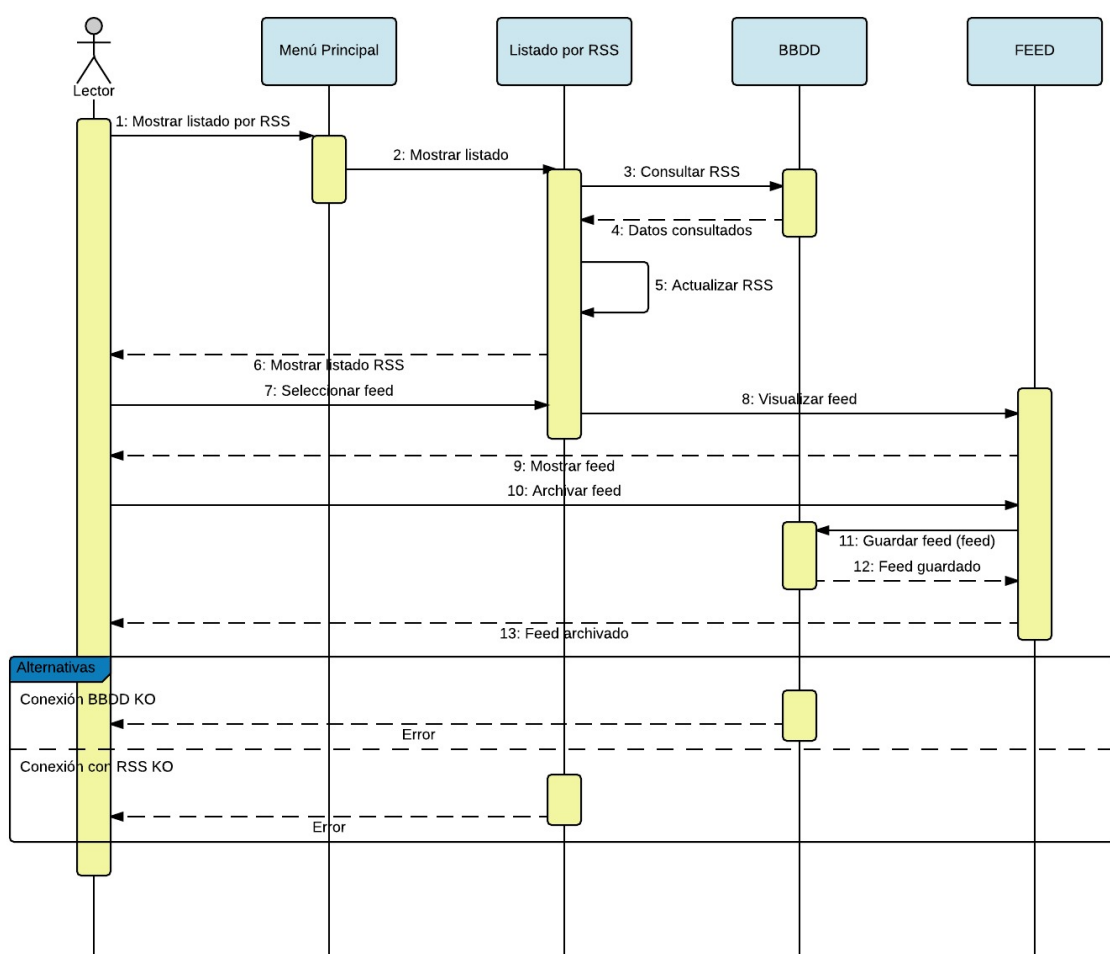


Ilustración 30: Diagrama de secuencia archivar feed

4.2.8. VISUALIZAR FEED EN SITIO WEB

Es este apartado se representa la secuencia de pasos que sigue un usuario para visualizar un feed en el sitio web que lo publica. Para ello accederá al menú principal, seleccionará la opción FEEDS, que mostrará la pantalla con el listado de feeds disponibles, finalmente se seleccionará del listado, el feed que interese visualizar y seleccionará la opción ver en sitio web.

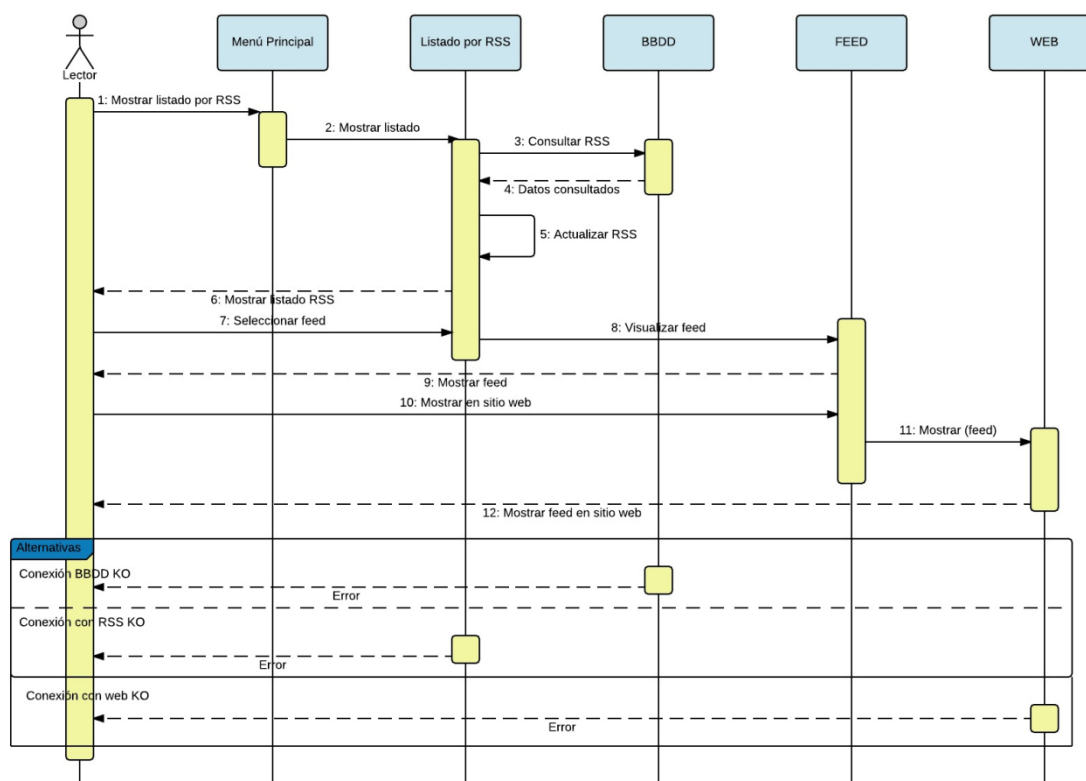


Ilustración 31: Diagrama de secuencia visualizar feed en sitio web

4.3. DIAGRAMAS DE CLASES

El diagrama de clases es un tipo de diagrama de estructura estática que describe la estructura de un sistema mostrando las clases del sistema, sus atributos, operaciones (o métodos), y las relaciones entre los objetos.

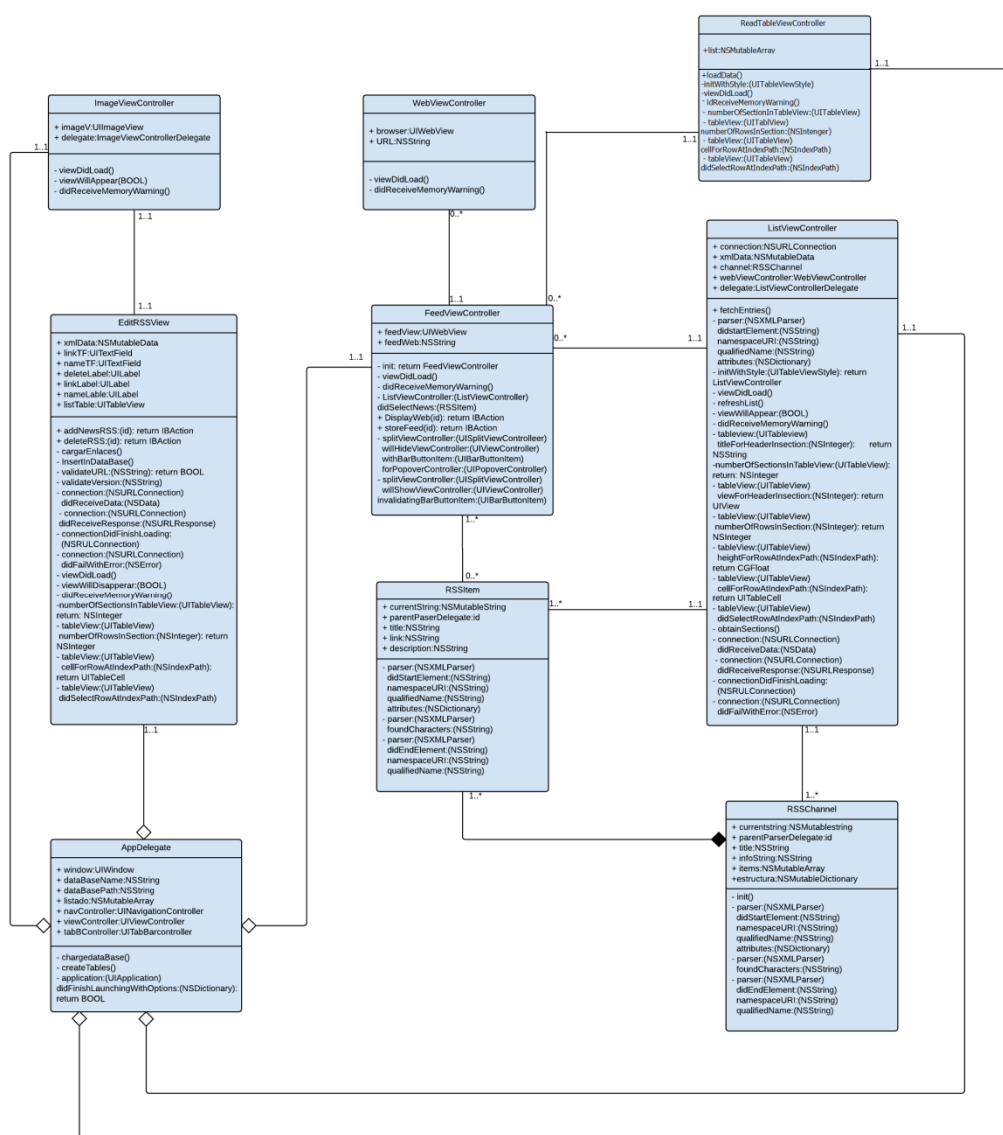


Ilustración 32: Diagrama de clases

Con el fin de completar una mejor comprensión del diagrama anterior, se describirá brevemente la función de cada una de las clases:

- **AppDelegate:** esta clase hereda de UIResponder, clase que se encarga de manejar eventos en iOS. Además implementa los métodos de delegado de UIApplicationDelegate. Principalmente será utilizada para iniciar la creación de otras vistas: ListViewController, FeedViewController, EditRSSView, ImageViewController.
- **EditRSSView:** esta clase hereda de UIViewController, clase que proporciona necesaria para gestionar las vistas de un app. Principalmente será utilizada para la presentación de la vista que permitirá gestionar los enlaces a servicios RSS.
- **ImageViewController:** esta clase hereda de UIViewController y principalmente será utilizada para presentar el logo de la app.
- **FeedViewController:** esta clase hereda de UIViewController y principalmente será utilizada para presentar la información de los feeds.
- **WebViewController:** esta clase hereda de UIViewController y principalmente será utilizada para abrir los feeds en el sitio web que los publica.
- **ListViewController:** esta clase hereda de UITableViewController, clase que se encarga de controlar las vistas de tablas. Principalmente será utilizada para listar los feeds publicados por los servicios RSS a los que el usuario está suscrito.
- **ReadTableViewController:** esta clase hereda de UITableViewController, clase que se encarga de controlar las vistas de tablas. Principalmente será utilizada para listar los feeds archivados por el usuario.
- **RSSChannel:** esta clase hereda de NSObject, clase raíz de la jerarquía de la mayoría de las clases de Objective-C. Principalmente será utilizada para parsear el XML publicado por los servicios RSS a los que el usuario está suscrito.
- **RSSItem:** esta clase hereda de NSObject, y principalmente será utilizada para parsear los item's de los que se compone el elemento "channel", del XML publicado por un servicio RSS.

4.3.1. ARQUITECTURA

A continuación se expone un diagrama de clases que representa de forma general la arquitectura diseñada para la aplicación.

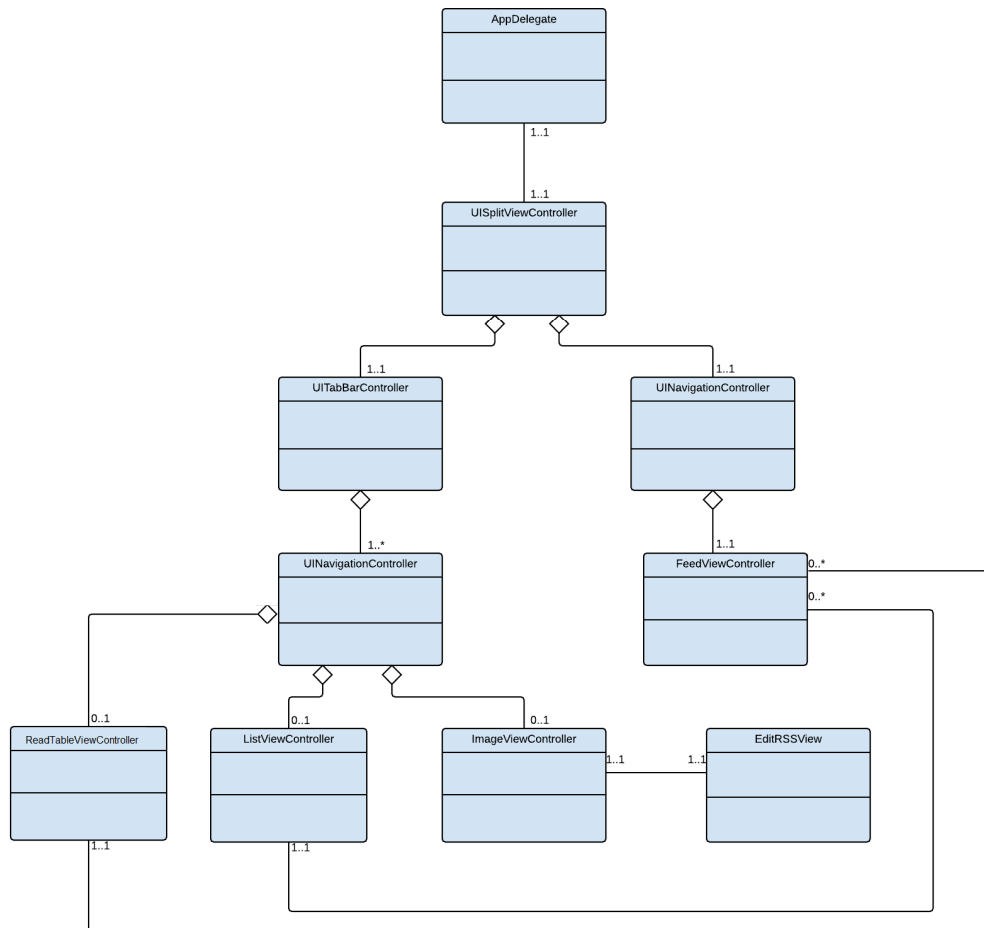


Ilustración 33: Diagrama arquitectura de clases

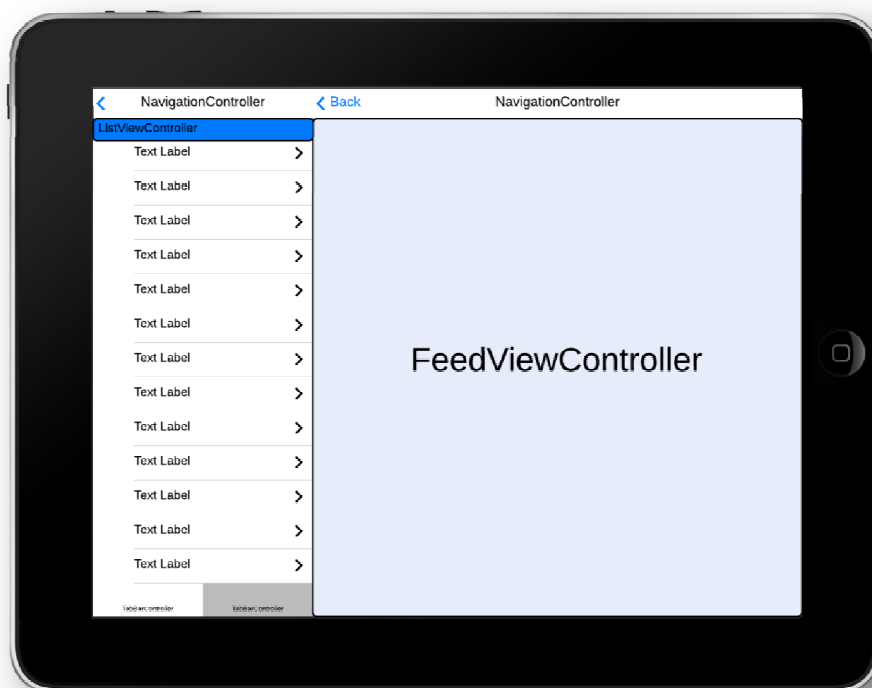


Ilustración 34: Maqueta interfaz I

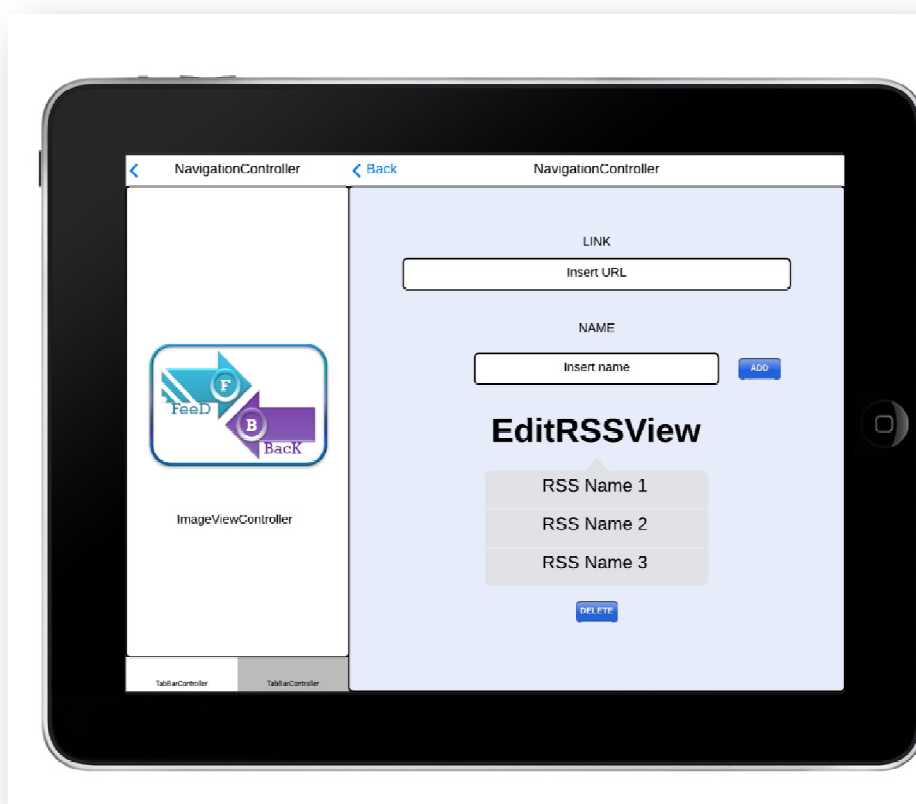


Ilustración 35: Maqueta interfaz II

CAPITULO 5: PRUEBAS

5.1. INTRODUCCIÓN

Este capítulo recoge la descripción de la batería de pruebas realizadas para la comprobación del correcto funcionamiento del sistema desarrollado.

Al margen de las pruebas unitarias realizadas paralelamente al desarrollo, de forma individualizada a cada uno de los componentes implementados, se han llevado a cabo un testeo exhaustivo de la integración de todos ellos, con el fin de asegurar de validar todos los requisitos especificados en el presente documento.

5.2. PRUEBAS INTEGRADAS.

A continuación se describen los campos que definen cada uno de los casos de prueba.

- **Identificador:** identifica de forma única cada caso de prueba. El formato utilizado es el siguiente "PI-XX", donde 'XX' se sustituirá por numeración ascendente, empezando por 01.
- **Descripción:** pequeña reseña acerca del caso de prueba a ejecutar.
- **Requisitos cubiertos:** identificadores de los requisitos que quedan estados a través de la prueba.
- **Actuación:** pasos a seguir para llevar a cabo el caso de prueba.
- **Errores posibles:** fallos susceptibles de ser encontrados a lo largo del caso de prueba.
- **Resultado esperado:** resultado que se espera de la ejecución del caso de prueba, se concluirá que el resultado es correcto si resultado de la prueba coincide con el aquí documentado.

PI-01	
Descripción	Suscripción a un enlace RSS
Requisitos cubiertos	RF-01
Actuación	<ol style="list-style-type: none"> 1. Seleccionar en el menú principal la opción "RSS". 2. Introducir los datos del enlace RSS: link y nombre. 3. Pulsar botón "ADD"
Errores posibles	<ol style="list-style-type: none"> A. La aplicación no almacena el link, aunque se hayan insertado correctamente los datos de entrada. B. No se dispone de conexión a internet. C. No es posible establecer conexión con el servidor del servicio RSS que se quiere suscribir.
Resultado esperado	Prueba satisfactoria, el nombre de la nueva suscripción aparece en la tabla de enlaces agregados de la misma pantalla.

Tabla 29: PI-01 Suscripción a un enlace RSS

PI-02	
Descripción	Borrado de un enlace RSS
Requisitos cubiertos	RF-02
Actuación	<ol style="list-style-type: none"> 1. Seleccionar en el menú principal la opción "RSS". 2. Seleccionar en la tabla de enlaces suscritos el nombre del enlace que desea borrar. 3. Automáticamente este nombre aparecerá en la etiqueta situada debajo de la tabla. 4. Pulsar botón "Delete."

Errores posibles	A. No se borra el enlace seleccionado.
Resultado esperado	Prueba satisfactoria, el nombre del enlace seleccionado desaparece de la tabla de enlaces agregados de la misma pantalla.

Tabla 30: PI-02 Borrado de un enlace RSS

PI-03	
Descripción	Validar servicio RSS a suscribir.
Requisitos cubiertos	RF-03
Actuación	<ol style="list-style-type: none"> 1. Seleccionar en el menú principal la opción "RSS". 2. Introducir los datos del enlace RSS: link sin formato URL y nombre. 3. Pulsar botón "ADD"
Errores posibles	A. La aplicación almacena los datos insertados.
Resultado esperado	Prueba satisfactoria, aparece un mensaje informado que el link insertado no es correcto.

Tabla 31: PI-03 Validar servicio RSS a suscribir

PI-04	
Descripción	Validar servicio RSS a suscribir.
Requisitos cubiertos	RF-03
Actuación	<ol style="list-style-type: none"> 1. Seleccionar en el menú principal la opción "RSS". 2. Introducir los datos del enlace RSS: link se un servicio RSS que no sea versión 2.0 y nombre. 3. Pulsar botón "ADD"
Errores posibles	<ol style="list-style-type: none"> A. La aplicación almacena los datos insertados. B. No se dispone de conexión a internet.

	C. No es posible establecer conexión con el servidor del servicio RSS que se quiere suscribir.
Resultado esperado	Prueba satisfactoria, aparece un mensaje informado que el servicio RSS no es versión 2.0.

Tabla 32: PI-04 Validar servicio RSS a suscribir

PI-05	
Descripción	Listar RSS suscritos
Requisitos cubiertos	RF-04
Actuación	<ol style="list-style-type: none"> 1. Seleccionar en el menú principal la opción "RSS". 2. Verificar los datos de la tabla que muestra la pantalla.
Errores posibles	A. La tabla de servicios suscritos aparece vacía.
Resultado esperado	Prueba satisfactoria, en la tabla de servicios suscritos aparece un listado con los links agregados.

Tabla 33: PI-05 Listar RSS suscritos

PI-06	
Descripción	Listar feeds por RSS
Requisitos cubiertos	RF-05
Actuación	1. Seleccionar en el menú principal la opción "FEEDS".
Errores posibles	<ol style="list-style-type: none"> A. No aparece listado en la parte izquierda de la pantalla. B. El listado aparece vacío a pesar de tener enlaces RSS suscritos. C. No se dispone de conexión a internet.

	D. No es posible establecer conexión con los servidores de los enlaces RSS suscritos.
Resultado esperado	Prueba satisfactoria, se visualiza una tabla en la parte izquierda de la pantalla, con los títulos de los feeds agrupados por secciones, con el nombre del RSS al que pertenecen.

Tabla 34: PI-06 Listar feeds a por RSS

PI-07	
Descripción	Visualizar feed
Requisitos cubiertos	RF-05, RF-06
Actuación	<ol style="list-style-type: none"> 1. Seleccionar en el menú principal la opción "FEEDS". 2. Seleccionar un feed del listado de feeds.
Errores posibles	<ol style="list-style-type: none"> A. El listado aparece vacío a pesar de tener enlaces RSS suscritos. B. No aparece información del feed a pesar de seleccionarlo en el listado. C. No se dispone de conexión a internet. D. No es posible establecer conexión con los servidores de los enlaces RSS suscritos.
Resultado esperado	Prueba satisfactoria, se visualiza la información del feed seleccionado en la parte derecha de la pantalla.

Tabla 35: PI-07 Visualizar feed

PI-08	
Descripción	Visualizar feed en sitio web
Requisitos cubiertos	RF-05, RF-06, RF-07
Actuación	<ol style="list-style-type: none"> 1. Seleccionar en el menú principal la opción "FEEDS". 2. Seleccionar un feed del listado de feeds. 3. Pulsar el botón browser.
Errores posibles	<ol style="list-style-type: none"> A. El listado aparece vacío a pesar de tener enlaces RSS suscritos. B. No aparece información del feed a pesar de seleccionarlo en el listado. C. No aparece el botón browser. D. No se dispone de conexión a internet. E. No es posible establecer conexión con los servidores de los enlaces RSS suscritos.
Resultado esperado	Prueba satisfactoria, se visualiza el feed en el sitio web que lo publica, en la parte derecha de la pantalla.

Tabla 36: PI-08 Visualizar feed en sitio web

PI-09	
Descripción	Archivar feed
Requisitos cubiertos	RF-05, RF-06, RF-08
Actuación	<ol style="list-style-type: none"> 1. Seleccionar en el menú principal la opción "FEEDS". 2. Seleccionar un feed del listado de feeds. 3. Pulsar el botón "archivar".
Errores posibles	<ol style="list-style-type: none"> A. El listado aparece vacío a pesar de tener enlaces RSS suscritos.

	<p>B. No aparece información del feed a pesar de seleccionarlo en el listado.</p> <p>C. No aparece el botón archivar.</p> <p>D. No se dispone de conexión a internet.</p> <p>No es posible establecer conexión con los servidores de los enlaces RSS suscritos.</p>
Resultado esperado	Prueba satisfactoria, se visualiza el feed archivado en el listado de feeds archivados.

Tabla 37: PI-09 Archivar feed

PI-10	
Descripción	Actualizar feeds
Requisitos cubiertos	RF-05, RF-09
Actuación	<ol style="list-style-type: none"> 1. Prueba satisfactoria, se visualiza el feed en el sitio web que lo publica, en la parte derecha de la pantalla. 2. Arrastrar el inicio la tabla del listado de feeds, situada en la parte izquierda de la pantalla, hacia abajo.
Errores posibles	<p>A. No aparece listado en la parte izquierda de la pantalla.</p> <p>B. El listado aparece vacío a pesar de tener enlaces RSS suscritos.</p> <p>C. No se dispone de conexión a internet.</p> <p>D. No es posible establecer conexión con los servidores de los enlaces RSS suscritos.</p> <p>E. No funciona el patrón pull to refresh.</p>
Resultado esperado	Prueba satisfactoria, e actualiza el listado de feeds.

Tabla 38: PI-10 Actualizar feeds

PI-11	
Descripción	Listar feeds archivados
Requisitos cubiertos	RF-10
Actuación	1. Seleccionar en el menú principal la opción "ARCHIVO".
Errores posibles	A. No aparece listado en la parte izquierda de la pantalla. B. El listado aparece vacío a pesar de tener feeds archivados.
Resultado esperado	Prueba satisfactoria, se visualiza una tabla en la parte izquierda de la pantalla, con los títulos de los feeds archivados.

Tabla 39: PI-11 Listar feeds archivados

PI-12	
Descripción	Marcar feed como no leído
Requisitos cubiertos	RF-06, RF-10, RF-11
Actuación	1. Seleccionar en el menú principal la opción "ARCHIVO". 2. Seleccionar un feed del listado de feeds. 3. Pulsar el botón "DELETE"
Errores posibles	A. No aparece listado en la parte izquierda de la pantalla. B. El listado aparece vacío a pesar de tener feeds archivados. C. No aparece información del feed a pesar de seleccionarlo en el listado. D. No desaparece el feed del listado.

Resultado esperado	Prueba satisfactoria, el feed desaparece del listado de feeds archivados.
---------------------------	---

Tabla 40: PI-12 Marcar feed como no leído

5.3. TRAZABILIDAD CASOS DE PRUEBA- REQUISITOS FUNCIONALES

	RF-01	RF-02	RF-03	RF-04	RF-05	RF-06	RF-07	RF-08	RF-09	RF-10	RF-11
PI-01	X										
PI-02		X									
PI-03			X								
PI-04			X								
PI-05				X							
PI-06					X						
PI-07					X	X					
PI-08					X	X	X				
PI-09					X	X		X			
PI-10					X				X		
PI-11										X	
PI-12						X				X	X

Tabla 41: Trazabilidad casos de prueba - requisitos funcionales

CAPITULO 6: PRESUPUESTO

6.1. INTRODUCCIÓN

En este apartado se presenta una estimación de los costes y duración derivados de la realización de este proyecto, desglosado en las siguientes partes:

- **Costes de personal:** costes del personal involucrado en la elaboración del proyecto.
- **Costes de material:** costes derivados de los componentes hardware y software necesarios para la elaboración del proyecto, teniendo en cuenta su amortización.
- **Otros costes directos:** costes que no se contemplan en ninguno de los apartados anteriores pero que son computables en el presupuesto total del proyecto.
- **Coste total:** suma de todos los costes parciales anteriormente indicados. Se presenta un resumen detallado de todos los costes incurridos.
- **Duración del proyecto:** muestra las fases en las que se ha dividido el proyecto, la duración de cada una de ellas, así como la fecha de inicio y la fecha de fin de cada una de ellas.

6.2. COSTES DE PERSONAL

En la elaboración del presente proyecto han intervenido las siguientes personas:

- Natalia Rodríguez González como gestor y desarrollador principal.

A continuación se muestra una tabla resumen de los costes especificados.

Nombre y Apellidos	Categoría	Horas totales (horas)	Coste persona/hora (euros)	TOTAL (euros)
Natalia Rodríguez González	Ingeniero semi sénior	632	15	9480
TOTAL				9480,00€

Tabla 42: Costes de personal

El coste de personal asciende a **9480 €**.

6.3. COSTES MATERIALES

Para el cálculo de los costes asociados a los materiales utilizados en la elaboración del presente proyecto hay que tener en cuenta la amortización de dichos materiales. Para ello se utilizará la siguiente fórmula:

$$A/B \cdot C \cdot D$$

- **A**= nº de meses desde la fecha de facturación en que el equipo es utilizado.
- **B**=periodo de amortización.
- **C**=coste del equipo (sin IVA)
- **D**=% del uso que se dedica al proyecto.

A continuación se muestra una tabla resumen de los costes especificados.

Descripción	Dedicación (meses)	Periodo de amortización (meses)	Coste (euros, sin IVA)	% de uso dedicado al proyecto D	Coste imputable (euros)
	A	B	C		
Ordenador portátil HP	3	48	600	50	18,75
Mac mini	3	48	659	100	41,19
Periféricos	3	48	250	100	15,63
Windows 8	3	48	99	50	3,09
Office 2013	3	24	117	50	7,31
Microsoft Project 2010	0,5	24	490	50	5,10
Curso on-line iOS	1	12	99	100	8,25
TOTAL					99,32

Tabla 43: Costes materiales

El coste total de materiales asciende a **99,32 €**.

6.4. OTROS COSTES DIRECTOS

Por último, se refleja una tabla con otros costes directos del proyecto y que deben tenerse también en cuenta para poder estimar el presupuesto del mismo.

Descripción	Costes directos (euros)
Material de oficina	5
Transporte	10
TOTAL	15

Tabla 44: Otros costes directos

El total de costes directos asciende a **15 €**

6.5. COSTE TOTAL

El coste total del proyecto es la suma de todos los costes parciales anteriormente indicados.

PRESUPUESTO PFC

COSTES DE PERSONAL

Nombre y Apellidos	Categoría	Horas totales (horas)	Coste persona/hora (€)	TOTAL (€)
Natalia Rodríguez González	Ingeniero semi sénior	632	15	9480
			TOTAL (€)	9480,00

COSTES MATERIALES

Descripción	Dedicación (meses) A	Periodo de amortización (meses) B	Coste (€ sin IVA) C	% de uso dedicado al proyecto D	Coste imputable (€)
Ordenador portátil HP	3,00	48	600	50	18,75
Mac mini	3,00	48	659	100	41,19
Periféricos	3,00	48	250	100	15,63
Windows 8	3,00	48	99	50	3,09
Office 2013	3,00	24	117	50	7,31
Microsoft Project 2010	0,50	24	490	50	5,10
Curso on-line iOS	1,00	12	99	100	8,25
			TOTAL (€)		99,32

OTROS COSTES DIRECTOS

Descripción	Coste (€)
Material de oficina	5
Transporte	10
TOTAL (€)	15

COSTES INDIRECTOS

Σ (costes parciales)	%
9594,32	20
TOTAL (€)	1918,86

RESUMEN PRESUPUESTO

COSTES DE PERSONAL	9480,00
COSTES MATERIALES	99,32
OTROS COSTES DIRECTOS	15,00
COSTES INDIRECTOS	1918,86

SUBTOTAL (€)	11513,19
IVA (21%)	2417,77

TOTAL PRESUPUESTO 13930,96 €

Ilustración 36: Presupuesto aplicación

El presupuesto total del proyecto asciende a la cantidad de **TRECE MIL NOVECIENTOS TREINTA EUROS CON NOVENTA Y SEIS CENTIMOS**. Dicho presupuesto arroja un beneficio de un **82%** y un margen de riesgo de **2033,19€**.

6.6. DURACIÓN DEL PROYECTO

En este apartado se detallará el diagrama de Gantt basado en las fases que incluye este proyecto. Para cada una de las subtareas que incluyen las fases, se mostrará la duración y el periodo de tiempo en el que han sido elaboradas.

- **Planificación:** se han definido los objetivos y el alcance del proyecto, junto con la planificación a seguir en el plan de acción establecido. Además, se ha realizado un estudio de las tecnologías necesarias para el correcto desarrollo del proyecto.
- **Análisis:** se han identificado los casos de uso necesarios para cubrir los requisitos especificados en el apartado anterior.
- **Diseño:** se ha definido la arquitectura técnica que se ha utilizado en la aplicación (split view controller) y se documentan y definen, apoyándonos en diagramas, las estructuras de datos, la estructura de programa y los detalles procedimentales para los casos de uso especificados en la fase anterior.
- **Implementación:** se ha realizado la aplicación para satisfacer los requisitos establecidos.
- **Pruebas:** se ha ejecutado el plan de pruebas necesario para verificar el correcto funcionamiento del software desarrollado.
- **Documentación:** se han realizado los documentos entregables necesarios para la entrega de este proyecto:
 - Memoria del PFC.
 - Presentación necesaria para la defensa del proyecto.
 - Manual de usuario de la aplicación desarrollada.

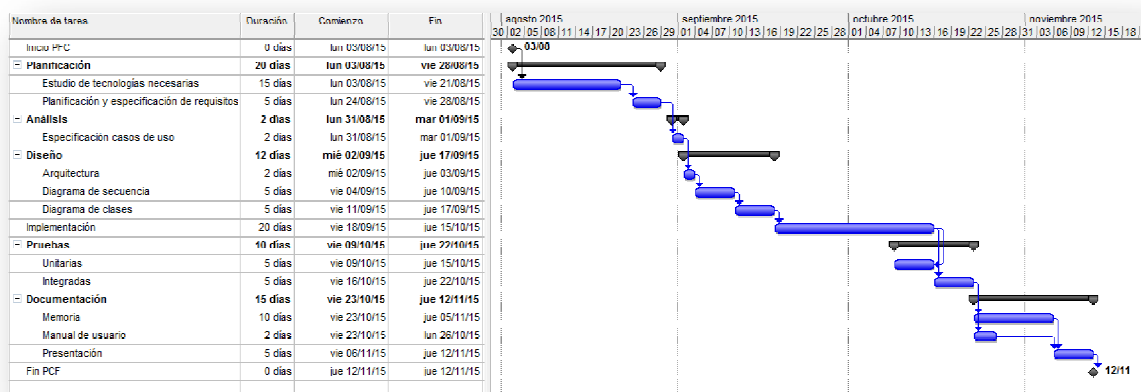


Ilustración 37: Diagrama Gantt

CAPITULO 7: CONCLUSIONES Y LINEAS FUTURAS

7.1. CONCLUSIONES

En el presente capítulo se pone el broche de oro al gran reto personal que ha supuesto la carrera de fondo realizada en mi paso por la UC3M, que representa un claro ejemplo de ¡¡querer es poder!!, y que no hay más límites que los que no ponemos a nosotros mismos.

La realización de este PFC no sólo ha servido para aplicar conocimientos adquiridos durante mi etapa universitaria, sino para aprender multitud de cosas nuevas y profundizar en un mundo tan puntero hoy en día como es el de las aplicaciones móviles, o como lo llaman los expertos “cultura de las apps móviles”. Un mercado con fuerte crecimiento y cuyo impacto en el mundo laboral lo estamos viendo día a día.

El teléfono móvil forma parte de nuestra vida y nuestro trabajo, y cada vez permite realizar más funciones y tareas propias de otros dispositivos. Por ello se me hace este mundo tan llamativo y diverso. Ofrece un amplio abanico de tipos de usuarios, plataformas, tipos de aplicaciones, temáticas y sectores.

Por ello quise aprovechar esta oportunidad para adéntrame en un mundo totalmente desconocido para mí, tanto a nivel usuario como desarrolladora, como es el mundo iOS. A pesar de las dificultades que conlleva aprender nuevas tecnologías, valoro de forma muy positiva el haber elegido esta tecnología, porque me ha aportado la visión del trabajo con apps.

Finalmente, una vez terminado el presente PFC se concluye que los resultados obtenidos son satisfactorios, ya que la aplicación desarrollada cumple con los objetivos impuestos al principio del desarrollo de la misma y a nivel personal termino con un reto personal, lo que empezó como un reto personal, puedo concluir que: misión cumplida.

7.2. LINEAS FUTURAS

El desarrollo de este PFC ha supuesto una primera versión de la aplicación, la cual contiene la funcionalidad indicada en los objetivos y que constituye las opciones básicas que el usuario necesita para poder utilizar la aplicación de la forma en que ésta ha sido pensada.

Es por ello que, al ser una primera versión de la aplicación, es interesante pensar en nuevas mejoras que en un futuro podría ser viable implementar en dicha aplicación, para enriquecer y construir una mejor aplicación que sea mucho más útil para los usuarios.

Algunas de las posibles nuevas mejoras que se podrían implementar en futuras versiones de la aplicación son las siguientes:

- **Categorización de los enlaces RSS:** se podría incluir una funcionalidad para agrupar los enlaces RSS en categorías, creadas por los propios usuarios, permitiendo una “personalización” de la interfaz, que el usuario pudiera elegir los colores e imágenes asociadas a las categorías creadas permitiendo así un entorno adaptable y personal.
- **Notificaciones:** se podría incluir un módulo de notificaciones, de tal forma que el usuario esté informado de cuando se publican nuevos feeds en los enlaces RSS en los que está suscrito.
- **Actualización automática de feeds:** además de la actualización manual, podría incluirse una funcionalidad que actualizara los feed de manera automática mediante jobs que se ejecutaran cada X tiempo, pudiendo aprovechar los campos “ttl, skipHours y skipDays” del servicio RSS suscrito.
- **Inclusión de nuevas versiones de RSS:** si en el futuro apareciese una nueva versión de RSS, sería necesario incluir una nueva funcionalidad que verifique dicha versión y sea capaz de interpretar los feeds publicados por esta.
- **Inclusión de un motor de búsqueda:** se podría incluir un motor de búsqueda especializado en servicios RSS, para facilitar la tarea de obtener el vínculos con el enlace del feed RSS, sin necesidad de que el usuario tenga que obtenerlos por fuera de la aplicación y gestionarlos de manera “manual”.

- **Integración con redes sociales:** las redes sociales son algo a tener en cuenta en la actualidad y muchas aplicaciones incorporan la posibilidad de que éstas se integren con las redes sociales más habituales, para que los usuarios muestren la actividad que realizan en la aplicación a través de las redes sociales que usan.

REFERENCIAS

- [1] La oreja de Van Gogh, "Nadie como tú" (2003).
- [2] Zack Snyder, "300" (2006)
- [3] "RSS softwaregarden", [Online]. Available: <http://rss.softwaregarden.com/aboutrss.html>
- [4] "Open Course Ware", [Online]. Available: http://ocw.uc3m.es/rss_feeds
- [5] "Bicimap", [Online]. Available: <http://bicimap.es/wordpress/?p=767>
- [6] "Developer mozilla", [Online]. Available: <https://developer.mozilla.org/es/docs/RSS/Versiones>
- [7] "Baquia", [Online]. Available: <http://www.baquia.com/emprendedores/los-5-mejores-lectores-de-feed-rss>
- [8] "Blogthinkbig", [Online]. Available: <http://blogthinkbig.com/lectores-de-feeds-smartphone/>
- [9] "NetMarketShare", [Online]. Available: <http://www.netmarketshare.com/>
- [10] "Blog ticsandroll", [Online]. Available: <http://blog.ticsandroll.es/actualidad/ios-y-su-arquitectura-interna-en-4-capas/>
- [11] Francisco Javier Honrubia López, "Programación de aplicaciones para Iphone y Ipad – Introducción a Objective-C". Available: www.aulamentor.es
- [12] Francisco Javier Honrubia López, "Programación de aplicaciones para Iphone y Ipad – Introducción a XCODE 4". Available: www.aulamentor.es
- [13] Francisco Javier Honrubia López, "Programación de aplicaciones para Iphone y Ipad – Patrón Modelo-Vista-Controlador". Available: www.aulamentor.es
- [14] Francisco Javier Honrubia López, "Programación de aplicaciones para Iphone y Ipad – Patrón Delegate". Available: www.aulamentor.es
- [15] Francisco Javier Honrubia López, "Programación de aplicaciones para Iphone y Ipad – Centro de notificaciones". Available: www.aulamentor.es
- [16] Francisco Javier Honrubia López, "Programación de aplicaciones para Iphone y Ipad – Target-Action". Available: www.aulamentor.es
- [17] Joe Conway and Aaron Hillegass, iOS Programming: The Big Nerd Ranch Guide, Third Edition (2012).



- [18] "Programación iOS para iPhone y iPad", [Online]. Available: www.udemy.com
- [19] "Apple Developer", [Online]. Available: <https://developer.apple.com>
- [20] "Wikipedia", [Online]. Available: www.wikipedia.org
- [21] "Lucidchart", [Online]. Available: www.lucidchart.com

ANEXO I: MANUAL DE USUARIO

A.1. INTRODUCCIÓN A FEEDBACK

FeedBack es una aplicación de las que se conoce como lector RSS o agregador RSS, la cual le permite obtener de forma rápida y precisa, las novedades producidas en sus sitios web favoritos, aprovechando las ventajas ofrecidas por los servicios RSS que estas publican.

Para comenzar a utilizar esta aplicación sólo hacen falta dos cosas:

1. Primero, que esa página web o blog que tanto le gusta, tenga disponible el servicio RSS, lo cual es bastante habitual y lo reconocerás porque tienen un logotipo naranja como el que se muestra a continuación:



Ilustración 38: Icono RSS II

2. Segundo, instalar en tu iPad FeedBack, donde deberás dar de alta aquellas páginas de las que deseas recibir noticias. Se trata de un proceso muy sencillo. Cuando te des de alta, podrás recibir directamente las noticias y nuevos artículos de aquellas páginas web que más te interesen y que hayas elegido incluir en tus "feeds", con la ventaja de tener todas las páginas web que te interesan reunidas en un único lugar.

A.2. CÓMO SUSCRIBIRSE A UN SERVICIO RSS

Inicialmente, como hemos comentado anteriormente es necesario verificar que tu sitio web favorito tiene disponible el servicio RSS, lo cual reconocerás por el icono



Ejemplos:

- **Diario 20minutos.es:**

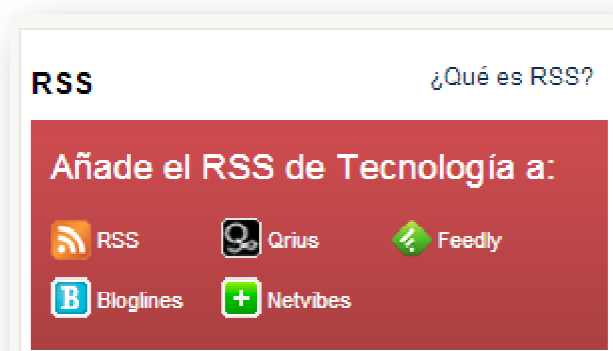


Ilustración 39: Identificación enlaces RSS 20minutos.es

- **Boletín Oficial UC3M:**

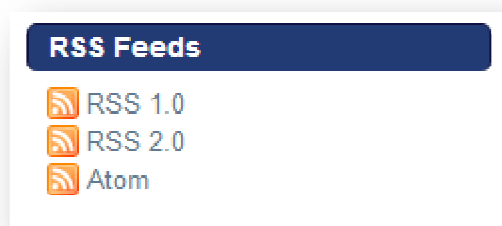


Ilustración 40: Identificación enlaces RSS boletín oficial UC3M

Pinchando en el icono identificativo de RSS obtendremos el link necesario para suscribirse al servicio RSS de nuestro sitio favorito.

➤ **Diario 20minutos.es:**

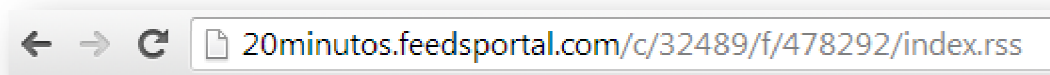


Ilustración 41: URL RSS 20minutos.es

➤ **Boletín Oficial UC3M:**

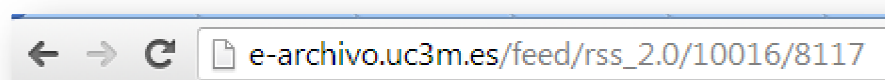


Ilustración 42: URL RSS boletín oficial UC3M

Una vez hemos obtenido/copiado el link deseado, abrimos nuestra aplicación FeedBack.

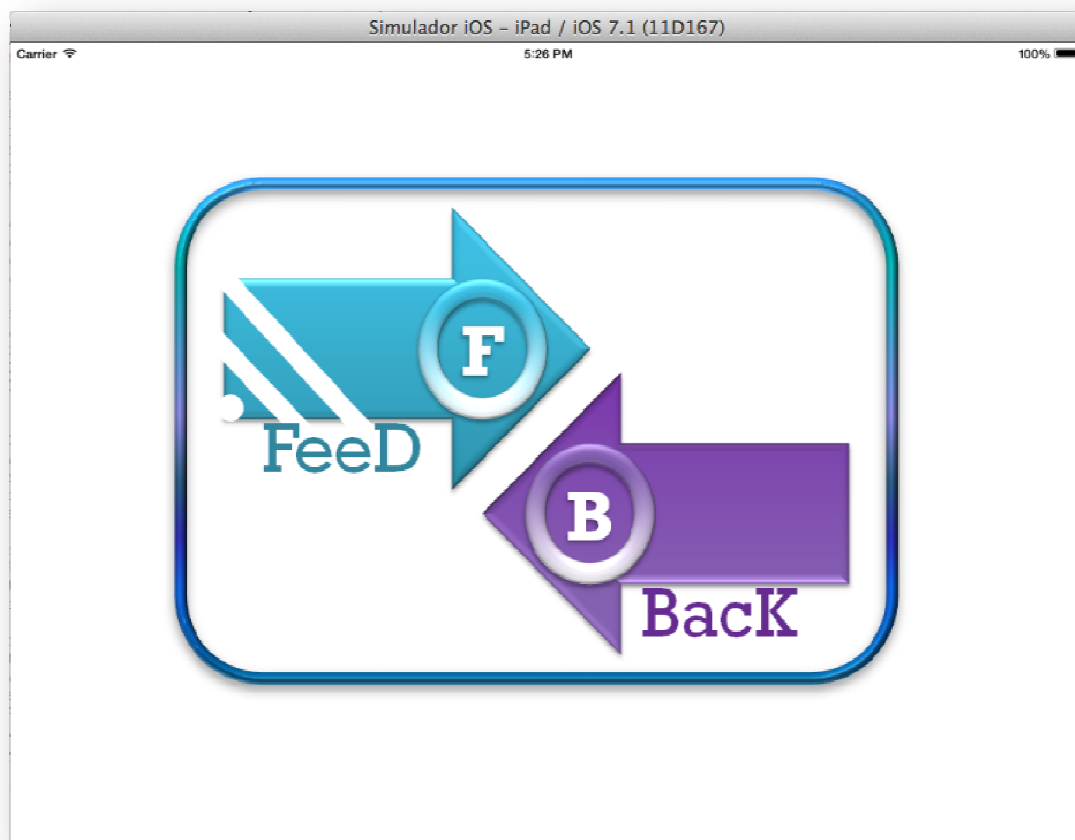


Ilustración 43: Pantalla inicio FeeDBack

Y nos dirigimos al menú



donde veremos el siguiente formulario:

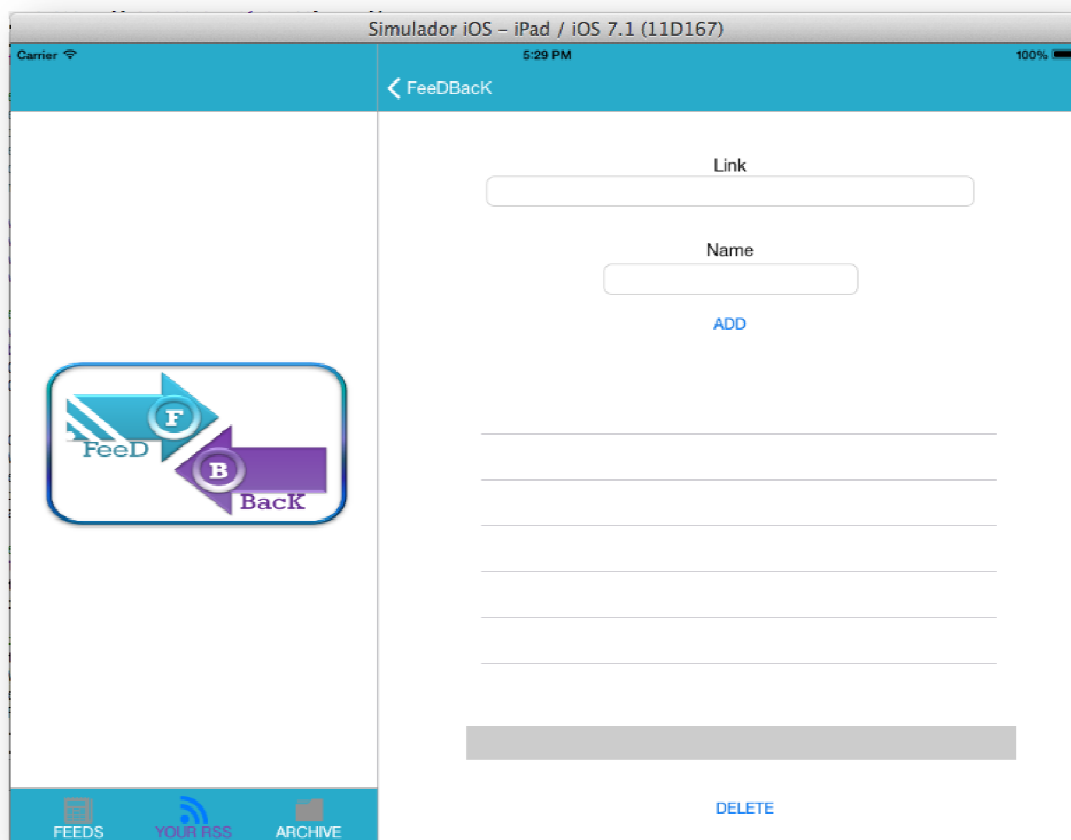


Ilustración 44: Ventana FeeDBack gestión enlaces RSS

Pegaremos en el campo “**Link**” el enlace que hemos copiado previamente, le asociaremos un **nombre** y pulsaremos el botón **ADD**. Si el link pegado es correcto y se verifica que es versión RSS 2.0, se actualizará la tabla con el nombre del link insertado. Eso es todo, ya estamos preparados para recibir toda la información actualizada de nuestro sitio web favorito.

Ilustración 45: Datos obligatorios alta RSS

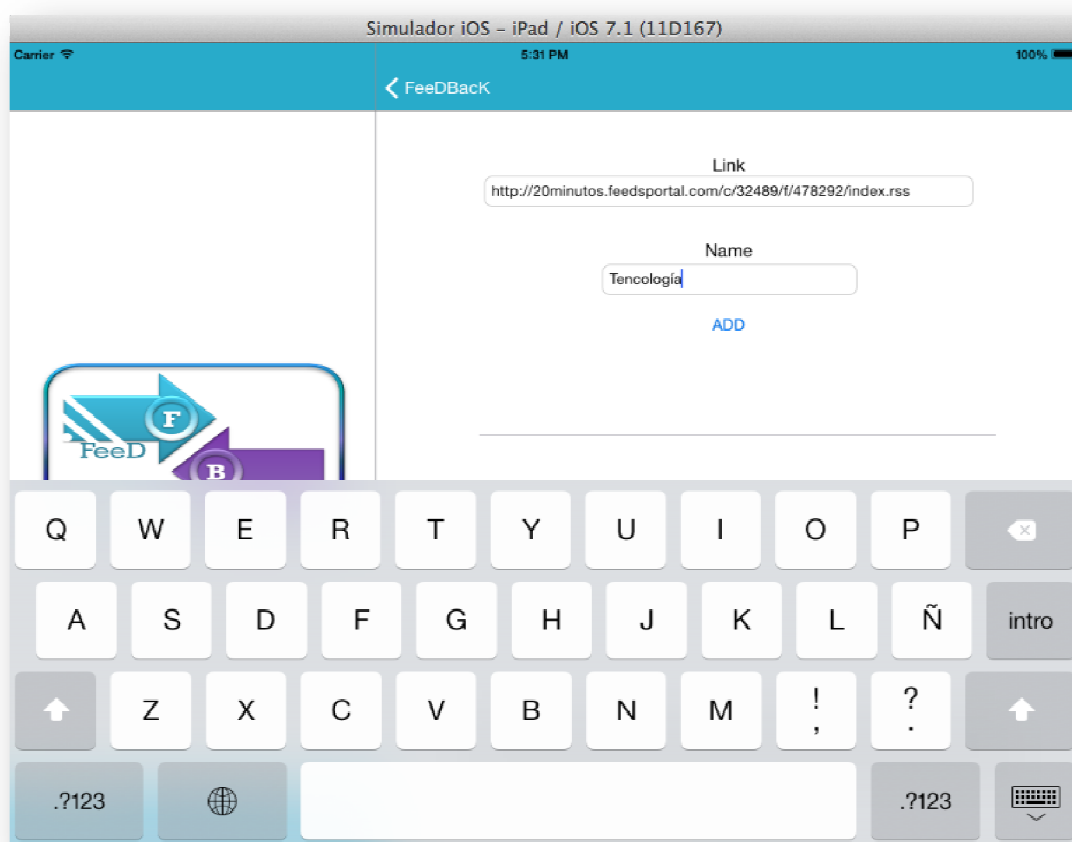


Ilustración 46: Operación alta servicio RSS

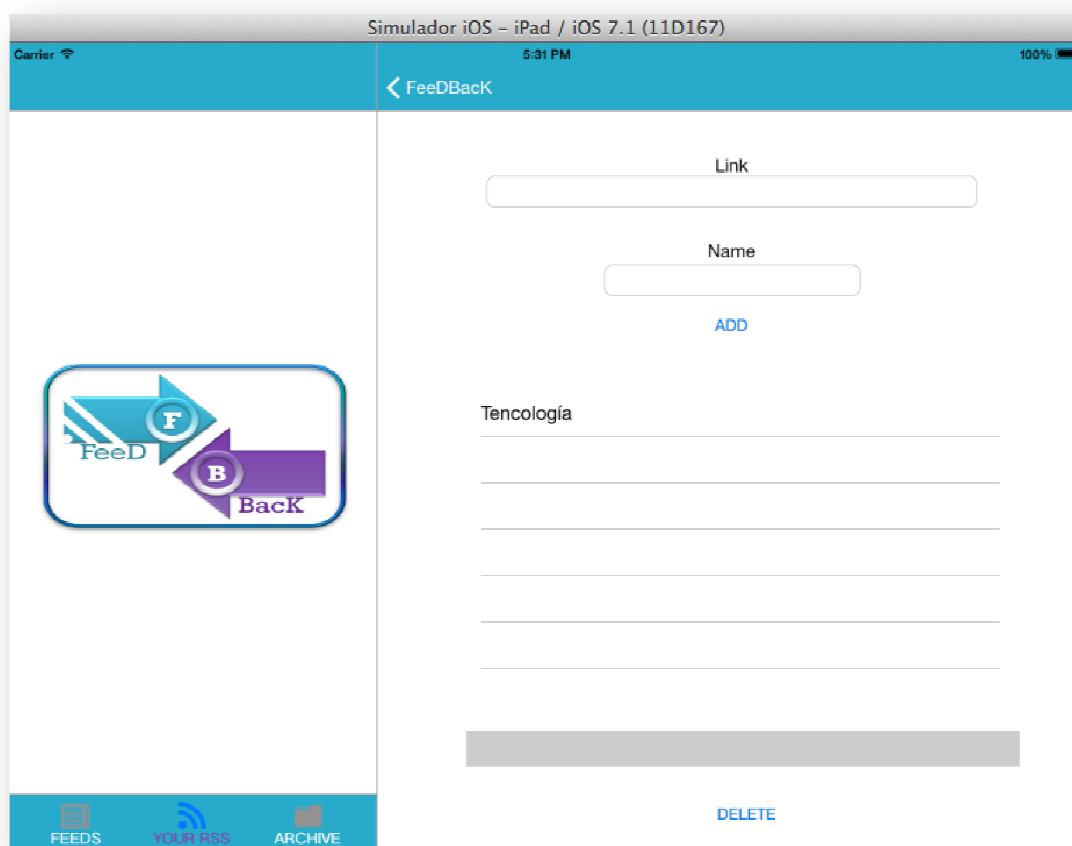


Ilustración 47: Alta servicio RSS realizada

En el caso de que el link no sea de la versión esperada (RSS 2.0) se visualizará un mensaje como el siguiente:

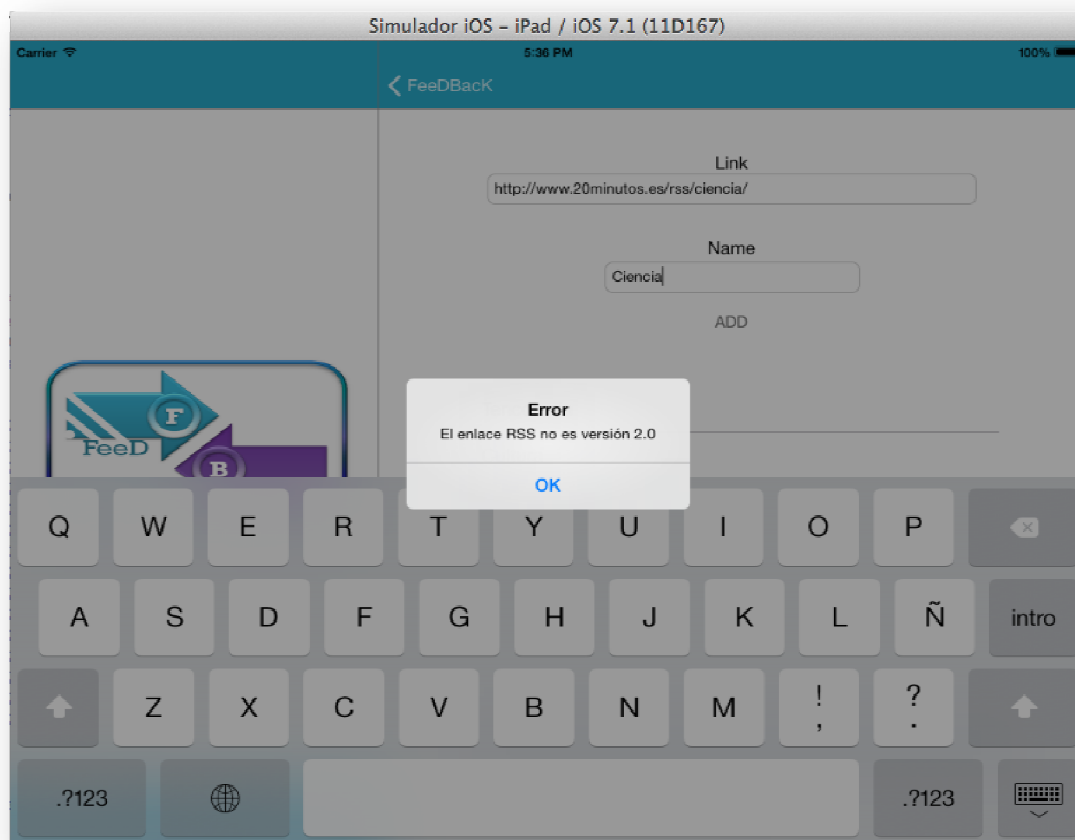


Ilustración 48: Error enlace RSS versión no soportada

A.3. CÓMO VISUALIZAR LOS FEEDS.

Una vez nos hemos suscrito a los servicios RSS de nuestros sitios web favoritos, visualizar la información publicada por estos, es muy sencillo. Nos dirigimos al


menú  y visualizaremos en la parte izquierda de la pantalla un listado de titulares agrupados por sitio web de publicación.



Ilustración 49: Listado feeds por RSS



Ilustración 50: Ventana listado/visualización feeds por RSS

Para ampliar información sobre alguno de estos titulares es tan sencillo como pulsar sobre él, y automáticamente aparecerá en la parte derecha de la pantalla la información ampliada asociada a ese feed.

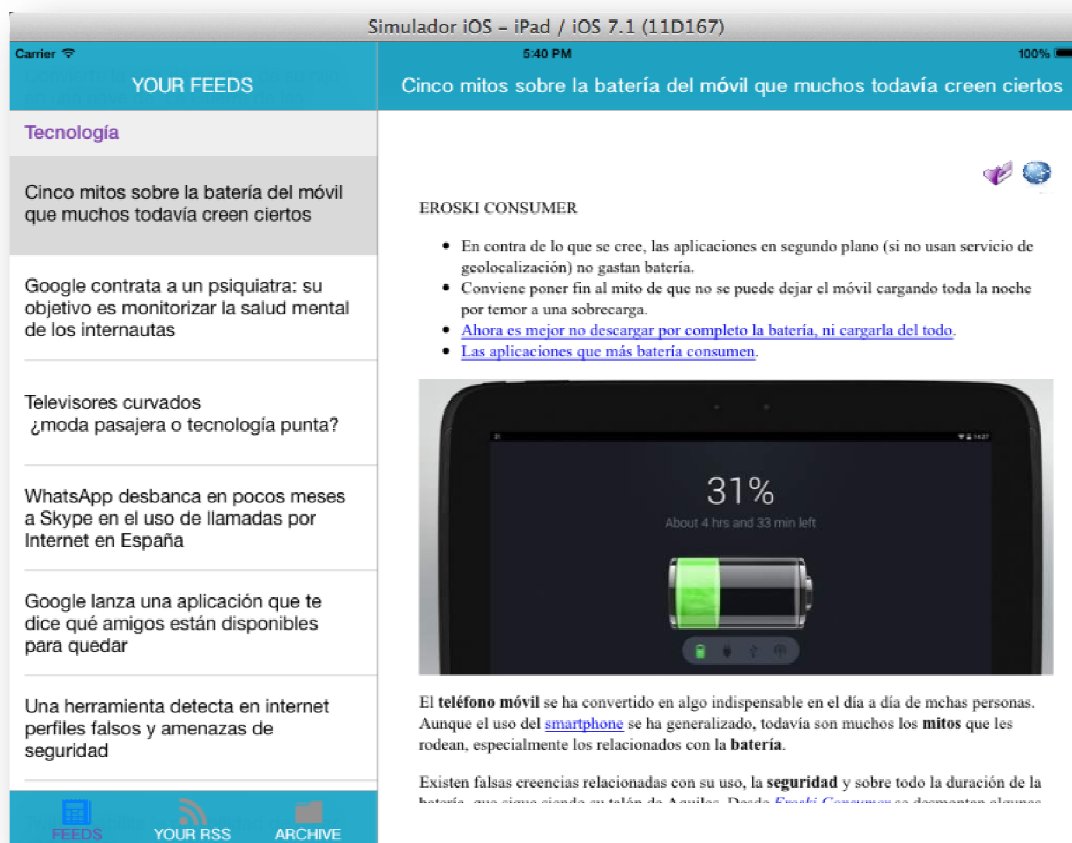



Ilustración 51: Ventana visualizar feed

Además, es posible visualizar la información del feed seleccionado en el sitio web que la publica dentro de la propia aplicación. Para ello bastará con pulsar el botón  que se encuentra dentro de la pantalla del feed.

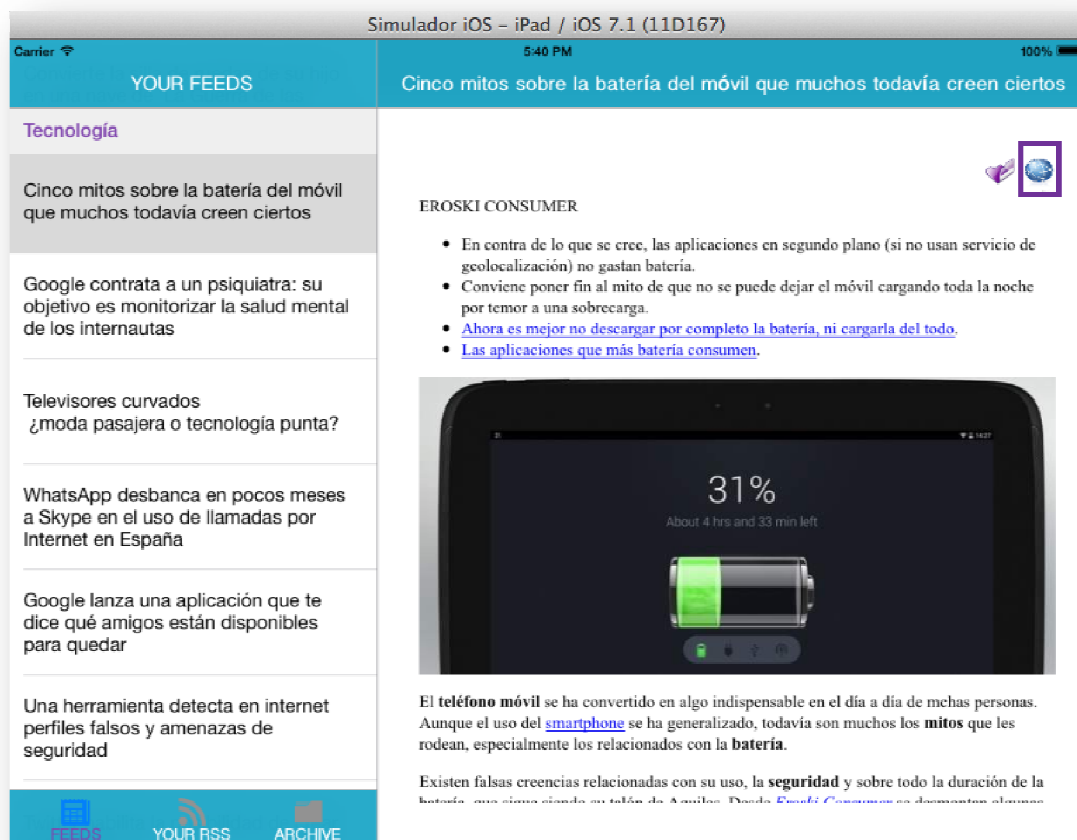


Ilustración 52: Botón para ver feed en sitio web

A.4. CÓMO ACTUALIZAR LOS FEEDS.

Siempre que deseemos obtener la información actualizada de nuestros servicios RSS suscritos deberemos dirigirnos al listado de feeds del menú "FEEDS" y arrastrar del inicio del listado hacia abajo, de esta forma el listado quedará actualizado automáticamente.



Ilustración 53: Actualizar feeds

A.5. CÓMO ARCHIVAR FEEDS.

FeedBack presenta la posibilidad de guardar el feed que estas visualizando para y disponer de él en local, de tal forma que en lugar de que el su ciclo de vida sea gestionado por el servicio RSS que lo publica, sea gestionado por el usuario de FeedBack, y disponer de dicha publicación hasta que lo considere necesario.



Para ellos bastará con pulsar el botón , que aparece en la ventana que presenta la información del feed.



Ilustración 54: Botón para archivar feed

A.6. CÓMO VISUALIZAR FEEDS ARCHIVADOS.

Después de archivar aquellos feeds que más nos interesan, recuperarlos es una tarea realmente sencilla. Nos dirigiremos al menú , y en la parte izquierda de la pantalla se visualizará un listado con los titulares de los feeds archivados.

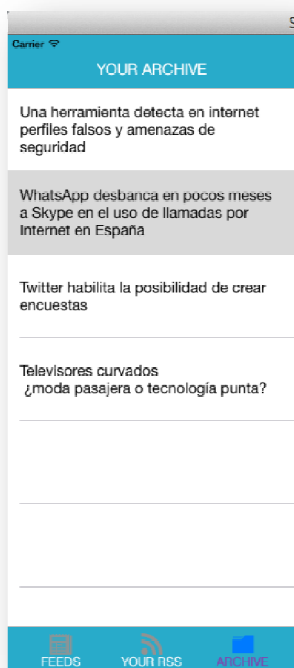


Ilustración 55: Listado feeds archivados

Para ampliar información sobre alguno de estos titulares basta con pulsar sobre él, y automáticamente aparecerá en la parte derecha de la pantalla la información ampliada asociada a ese feed.

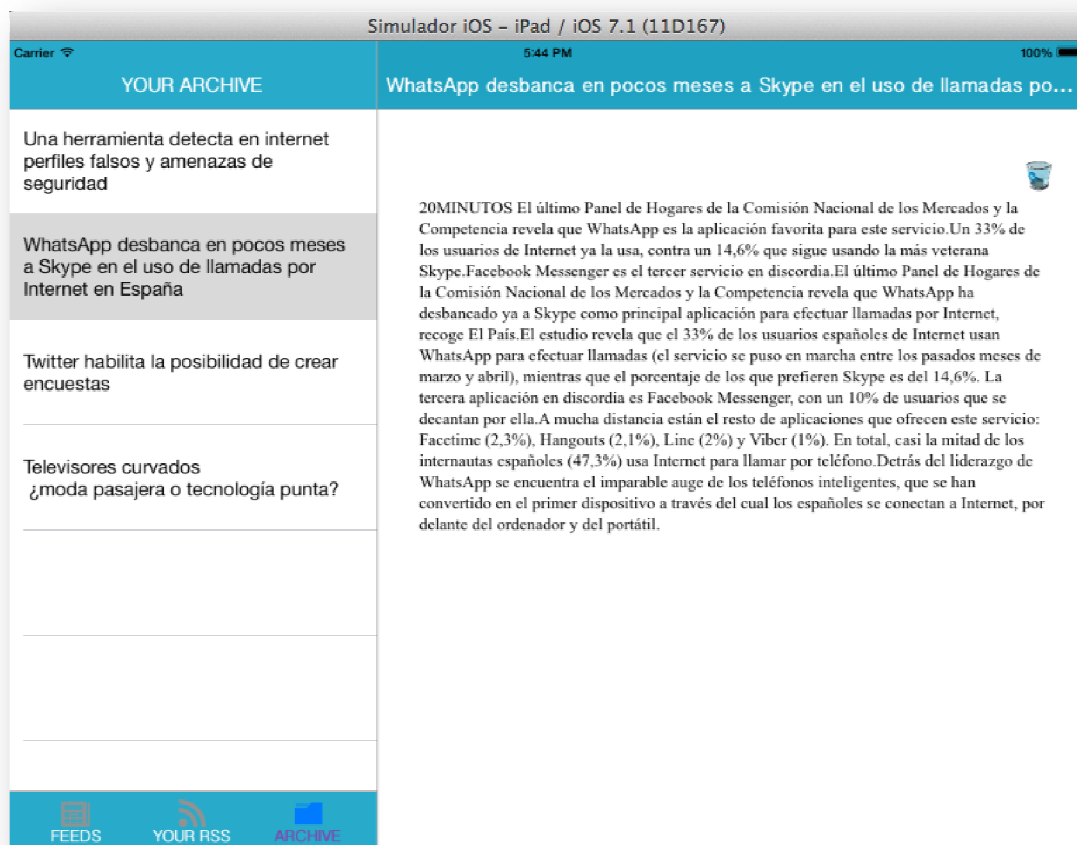



Ilustración 56: Ventana listado/visualización feeds archivados

Si en algún momento, alguno de los feeds de este listado deja de interesarnos, bastará con pulsar el botón  para eliminarlo.

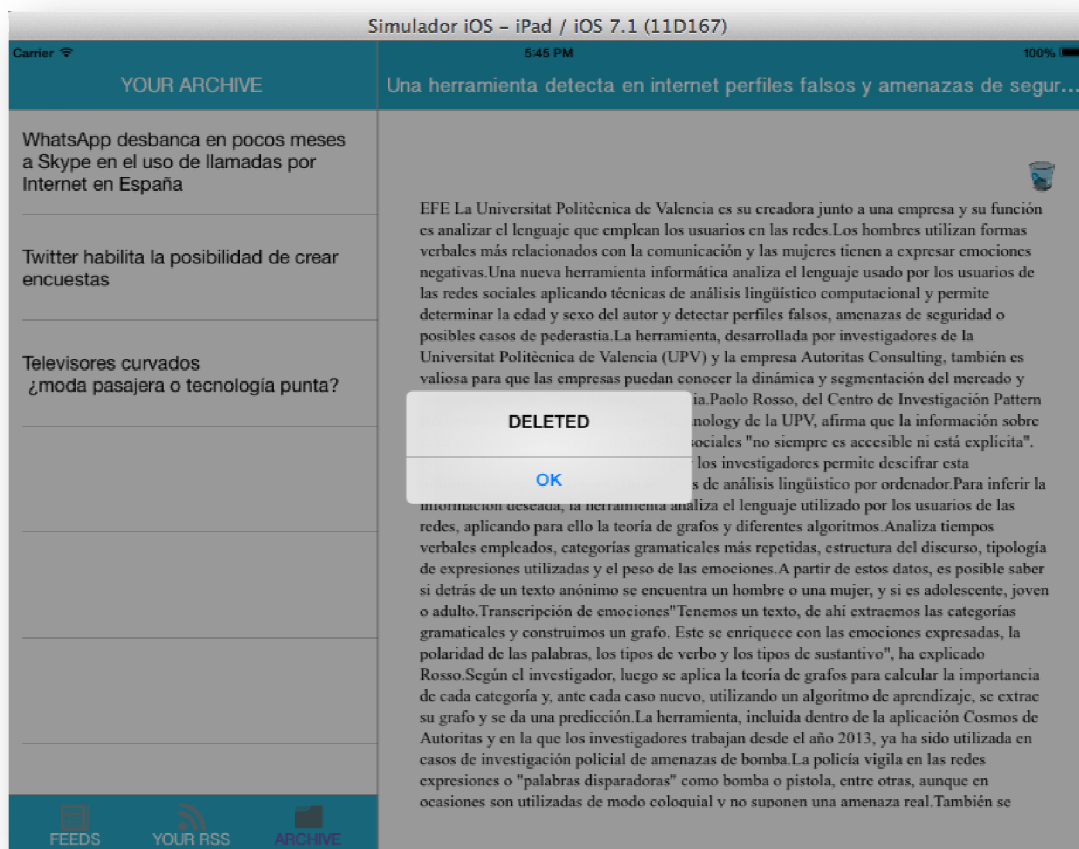


Ilustración 57: Borrar feed archivado

A.7. VISUALIZAR MENÚ

En el caso de querer visualizar la aplicación con el dispositivo en posición vertical, será posible acceder al menú, a través el del botón **SHOW MENU**.



Ilustración 58: Visualizar menú principal